AD-A053 618 PARKER (LEE W) INC CONCORD MASS F/G 20/3
POTENTIAL BARRIERS AND ASYMMETRIC SHEATHS DUE TO DIFFERENTIAL C--ETC(U)
JAN 78 L W PARKER F19628-76-C-0248
UNCLASSIFIED AFGL-TR-78-0045 NL

| OF |

AD
A053 618

END
DATE
FILMED
6-78
DDC

AD A053618

AD No.
DDC FILE COPY

AFGL-TR-78-0045

POTENTIAL BARRIERS AND ASYMMETRIC SHEATHS DUE TO DIFFERENTIAL CHARGING OF NONCONDUCTING SPACECRAFT

Lee W. Parker

Lee W. Parker, Inc.
252 Lexington Road
Concord, Massachusetts 01742

10 January 1978

Final Report

1 July 1976 through 30 September 1977

Approved for public release; distribution unlimited

AIR FORCE GEOPHYSICS LABORATORY
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
HANSCOM AFB, MASSACHUSETTS 01731

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

19 REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER 18 AFGL-TR-78-0045	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER 9
4. TITLE (and Subtitle) 6 POTENTIAL BARRIERS AND ASYMMETRIC SHEATHS DUE TO DIFFERENTIAL CHARGING OF NONCONDUCTING SPACECRAFT,		5. TYPE OF REPORT & PERIOD COVERED Final Report. 1 July 1976 — 30 Sept 1977
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) 10 Lee W. Parker		8. CONTRACT OR GRANT NUMBER(s) 15 F19628-76-C-0248 NEW
9. PERFORMING ORGANIZATION NAME AND ADDRESS Lee W. Parker, Inc. 252 Lexington Road Concord, Massachusetts 01742		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 62101F 16 76610701 17 07
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Geophysics Laboratory Hanscom AFB, Massachusetts 01731 Monitor/A. G. Rubin/PHG		12. REPORT DATE 11 10 January 1978
		13. NUMBER OF PAGES 56
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) 12 57p.		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

spacecraft charging	spacecraft sheaths	computer plasma
differential charging	spacecraft wakes	plasma simulation
plasma flows	potential barriers	kinetic theory
photoelectron emission	equilibrium potential	charged-particle orbits

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The differential charging of a nonconducting spacecraft is modeled numerically by following charged-particle trajectories in a self-consistent space-charge-less sheath. In the presence of a plasma flow, but independent of any photoelectric or secondary emission, a potential difference between the front and wake surfaces of the spacecraft is generated, resulting in an asymmetric sheath and in the creation of a potential barrier for electrons.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

391 809

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

The potential difference can amount to volts in the ionosphere, and kilovolts in the solar wind, that is, large compared with the potentials typically generated by photoemission alone. As in the more familiar case of photoelectric charging, the asymmetric sheath and potential barrier produced by the plasma flow can lead to erroneous interpretations of experiments measuring space electric fields and low-energy particle spectra. In an example of photoelectric emission, a sunlit area on an otherwise dark surface becomes positively charged by the emission, and is found to acquire a potential more than twice the emitted energy relative to the dark surface. This effect is associated with the physics of the "terminator" bounding the sunlit and dark areas.

1. INTRODUCTION

Differential spacecraft charging takes place when the spacecraft surface is partly or entirely insulating and the charged-particle fluxes vary from point to point over the surface. In the relatively familiar case of photoelectric emission from a sunlit insulated area (cf. Grard, 1973), due to electrons escaping from it the sunlit area tends to become positively charged relative to the surrounding dark areas. Another mechanism of differential charging, which is less familiar and appears to have been discussed little if at all in the literature, is that due to the relative motion between a nonconducting spacecraft and the external plasma (e.g., a spacecraft in the ionosphere or in the solar wind). The fluxes of ambient ions and electrons on the wake surface are not the same as on the front surface. For high velocities of relative motion compared with the mean ion thermal velocity, whether this occurs in the ionosphere (due principally to spacecraft motion) or in the solar wind (due principally to plasma motion), there is a significant differential in the ion fluxes, but a negligible differential for the electrons. Since the net current density must vanish locally at each surface point in the steady state, this plasma-flow effect leads to a larger negative equilibrium potential on the wake surface than on the front surface. If there is photoemission as well on the front surface (as in the solar wind), this differential charging is enhanced. The present report is concerned with these effects. As shown below, differences between the front and wake surface potentials amounting to many kT/e (where T is the temperature, k is Boltzmann's constant, and e is the electron charge), together with a potential barrier for electrons, can be generated by the motion. The potential differential may be expected to be of the order of volts (tens of kT) in the ionosphere, and of the order of a kilovolt (100 kT) in the solar wind. In the ionosphere and solar wind, this differential can therefore be much larger than that generated by photoemission alone. In the magnetosphere, however, the plasma-flow effect is relatively weak.

As of the mid-1960's there was already a considerable literature on the subject of estimating spacecraft potentials, using simplified models

without considering trajectories and assuming perfectly conducting spacecraft (see reviews by Brundin, 1963; Whipple, 1965; Samir and Willmore, 1966). These relatively crude models for estimating spacecraft potentials assumed either (a) very thin sheaths such that the "planar approximation" could be used, or (b) very thick sheaths (for small objects in the ionosphere) but with radial symmetry so that the simple Langmuir theory could be used. The crude estimates were not concerned with differential charging; they sufficed for treating effects where the detailed structure and asymmetry of the sheath (potential barriers, for example) were not considered important.

From the mid-1960's onward, spacecraft have increasingly sampled the magnetosphere and solar wind, where the spacecraft conditions are altered in several ways important for differential charging. First, there is an increase of sheath thickness to the order of the spacecraft dimensions, as opposed to the thin sheaths encountered in the ionosphere; this thickness is governed essentially by photoelectrons and secondary electrons from the surface, the space plasma contribution being typically relatively weak. A second circumstance is that many spacecraft surfaces are partly or entirely nonconducting (e.g., composed of glass-covered photocells or insulating thermal control blankets), which becomes important when the sheath is thick. A third circumstance is that nonuniform particle fluxes occur over the spacecraft surfaces, e.g., due to photoelectron and secondary emission and to plasma flows. The combination of the foregoing circumstances leads to differential charging of the spacecraft surfaces, which can have deleterious effects as follows.

If the differential charging is severe enough, spacecraft malfunctions can occur due to electrical discharges on the insulating surfaces (Fredricks and Scarf, 1973; Rosen, 1976). The appearance of hot magnetospheric plasmas in the kilovolt temperature range impacting on the dark surface, combined with photoelectric emission on the sunlit surface, make possible intense differential charging such that tens-of-kilovolts differentials can appear (see spacecraft charging symposia by Grard, 1973; Rosen, 1976; Pike and Lovell, 1977).

Even if the differential charging is not so strong, say no more than tens to hundreds of volts of differential, it can interfere with measurements of, say, weak ambient electric fields or low-energy particle spectra (Grard, 1973; Whipple and Parker, 1969). An interesting feature of differential charging as it affects low-energy electron measurements is that it can create electron potential barriers which can return emitted photoelectrons or secondary electrons to the surface and lead to erroneous interpretations of the data (Fahleson, 1973). This type of electron potential barrier is distinct from and should not be confused with the more familiar space-charge potential minimum of the order of a volt which can be produced by emitted-electron space charge and is not due to differential charging. The space-charge potential minimum has been studied theoretically by Soop (1972, 1973), Schröder (1973), Parker (1976b), Whipple (1976a), and Rothwell et al. (1977); it can, however, be less important than the barriers produced by differential charging effects. Discussing a well-documented experiment on the ATS-6 geosynchronous satellite, Whipple (1976b) infers that photoelectrons and secondary electrons from the spacecraft surfaces are reflected from a potential barrier which is much too large to be due to space charge but must be associated with some kind of differential charging (Whipple, 1976ab). A similar potential barrier due to differential charge is that produced artificially by an attractive electron trap mounted on a repulsive spacecraft, which can cause secondary-emission currents to be incorrectly interpreted as plasma currents (Whipple and Parker, 1969). In the present report, a potential differential and a potential barrier are shown to be producible by a plasma flow. This can lead to difficulties of interpretation of solar wind measurements such as those of Rosenbauer (1973), and may be responsible for singularities observed by photoelectron detectors in the ionosphere (W. K. Peterson, private communication, 1977).

The procedure for theoretically predicting sheath asymmetries and potential barriers is generally complicated in that it requires particle trajectory calculations as well as a three-dimensional sheath description for a realistic treatment (Parker, 1970, 1973, 1976a, 1977; Parker and Whipple, 1967, 1970; and appendix of this report).

In this report we present results of sample calculations of differential charging due to both plasma flow and photoemission, primarily addressing the asymmetric sheath and potential barrier produced by the plasma flow. These results may be considered preliminary, because the photoemission is considered separately rather than simultaneously. However, the differential charging is enhanced by photoelectric emission on the front surface. Space charge is neglected compared with surface charges as sources of the field; this neglect has been shown to be justified (Soop, 1973). Hence, while the predictions may not be quantitative for an actual spacecraft, they are conservative and indicate what may be expected: (a) in the ionosphere for small insulated objects, small meteoroids, or small parts of a spacecraft (e.g., a painted antenna) located within the wake region of a moving spacecraft, and (b) in the solar wind for an entire spacecraft.

In the example of photoelectric emission, a sunlit area on an otherwise dark surface becomes positively charged by emission, and is found to acquire a potential more than twice the emitted energy. This effect is associated with the physics of the "terminator" bounding the sunlit and dark areas.

In Section 2 we indicate briefly the nature of the numerical methods used, which are presented in more detail in Appendix A.

Results are discussed in Section 3.

A listing of the computer program used is given in Appendix B.

2. NUMERICAL APPROACH

The model chosen to represent a spacecraft is a truncated cylinder of approximately equal height and diameter, as shown in Fig. 1 where it is called (for brevity) a "pillbox." It is assumed that the spacecraft surface charge and the electric field around the spacecraft are axially symmetric, and that the electric field in space is given by the solution of Laplace's equation on a grid of points in r-z space (including the pillbox-shaped spacecraft). This means that when the potential distribution on the grid points has been computed (cf. Parker and Whipple, 1970), the potential and electric field at any point in space may be obtained by interpolation. This allows ion and electron trajectories to be computed within the region of space spanned by the grid. The outer boundary of the grid represents "infinity." The method used for computing the field allows the use of variable grid intervals (see Parker, 1976a, 1977; Parker and Whipple, 1970; and the appendix of this report).

The normal component of the ion or electron flux may be computed at any point on the pillbox surface by evaluating a triple integral in velocity space, and following trajectories backward in time to determine their origin and therefore the value of the integrand. This represents the "inside-out" method originated by Parker (1964). Details are given by Parker and Whipple (1970), and more generally by Parker (1976a, 1977) and in the appendix of this report. An alternative approach is the outside-in method (see appendix). For the calculations to be discussed, the conditions at the spacecraft surface are represented by a discrete set of grid points and associated surface areas, as illustrated (by 12 points) in Fig. 2.

For the present purposes, the differential potential and charge distributions on the spacecraft surface may be determined by two different approaches. In one approach we may determine the potential at each local grid point by a relaxation process until the net current density is zero. This involves "cutting-and-trying", whereby the surface potentials (12 values as illustrated in Fig. 2) are first given assumed values, and later successively corrected in accord with the signs and magnitudes of the resulting set of net current densities. The surface potentials represent

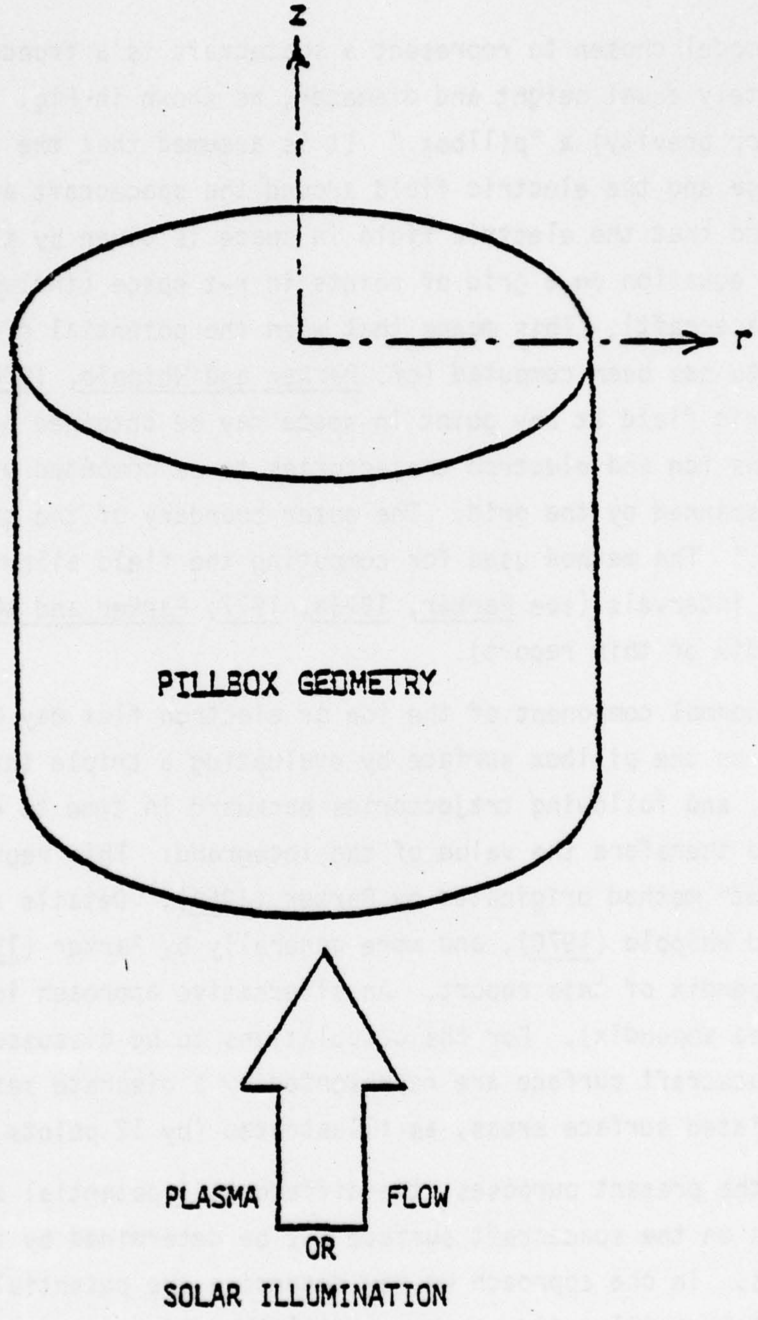

FIG. 1. SPACECRAFT AND PLASMA-FLOW OR SOLAR-ILLUMINATION GEOMETRY

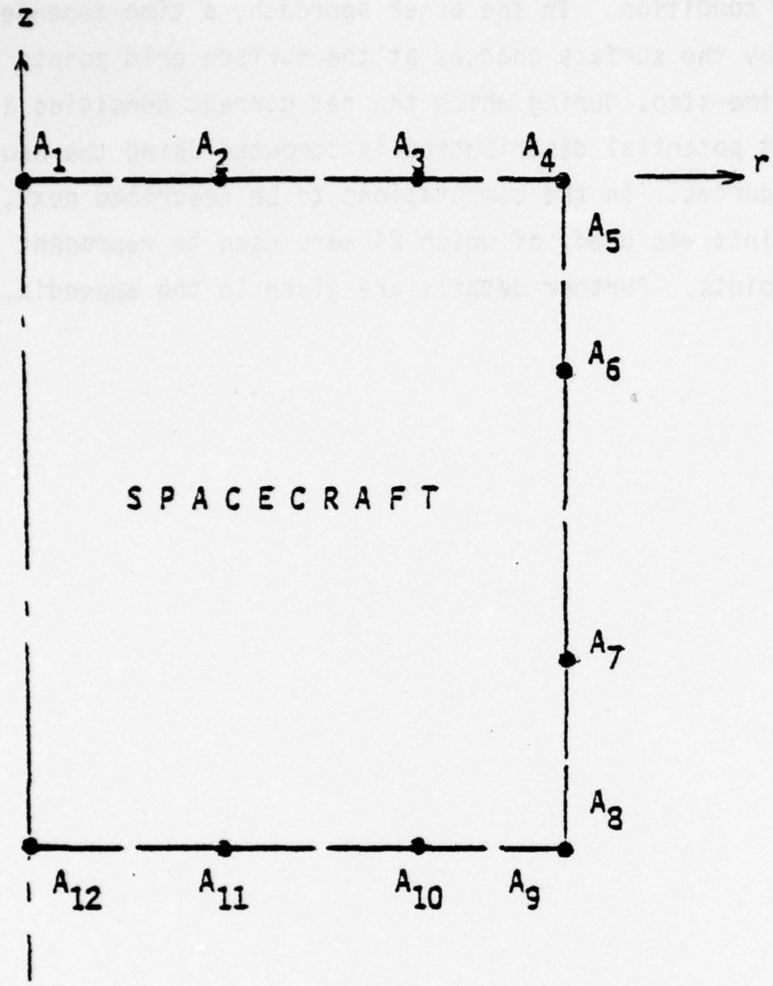

FIG. 2 - SURFACE AREAS ASSOCIATED WITH SURFACE POINTS ON SPACECRAFT

the "inner" boundary condition. In the other approach, a time-dependent method is used whereby the surface charges at the surface grid points are updated after each time-step, during which the net current densities are calculated. The next potential distribution is computed using the new surface charges as sources. In the computations to be described next, a grid of about 400 points was used, of which 24 were used to represent the spacecraft surface points. Further details are given in the appendix.

3. SAMPLE SOLUTIONS

A. Plasma Flow

In one of the problems treated here, we assume the nonconducting spacecraft to be in a flowing plasma, with the plasma flow along the axis, from the bottom (front region) toward the top (wake region) in Fig. 1. The plasma is taken to be ionized hydrogen and is assumed to have a velocity of flow four times larger than the most probable ion thermal velocity (ion "Mach number" = 4). Since the unperturbed ion flux to the wake surface is about 9 orders of magnitude smaller than the corresponding ion flux to the front surface, and since the electron fluxes are about the same to the front and wake surfaces, there will be a significant differential between the equilibrium potentials at the front and wake surfaces (see below). The effects of photoemission for the pillbox geometry are treated in the next section.

Figure 3 shows equipotential contours around the spacecraft, obtained by numerical solution, labeled by encircled numbers representing dimensionless values of the potential (in units of kT/e, where T is the plasma temperature). The errors in the solution shown are estimated to be under 10 percent. These potentials are obtained from Laplace's equation, where the surface potentials are obtained by the relaxation method discussed in the appendix, under the requirement of zero net current density at all surface points.

There are three regions of characteristic behavior of the potential, the "top" or "wake", the "side", and the "bottom" or "front". In the top region, the potentials are of the order of -10kT/e. This large negative value (about -17kT/e at the surface) is associated with the reduction in ion flux due to the flow. In the side region, the potentials are of the order of -3kT/e; this is essentially the order of the equilibrium potential when there is no flow ($\sim -\ln\sqrt{m_i/m_e}\ kT/e$). In the bottom region, the potentials are of the order of -kT/e, i.e., less negative than the side, because of the enhancement of the ion flux due to the flow. (Adding photoemission here would make the front potential still less negative.) The surface

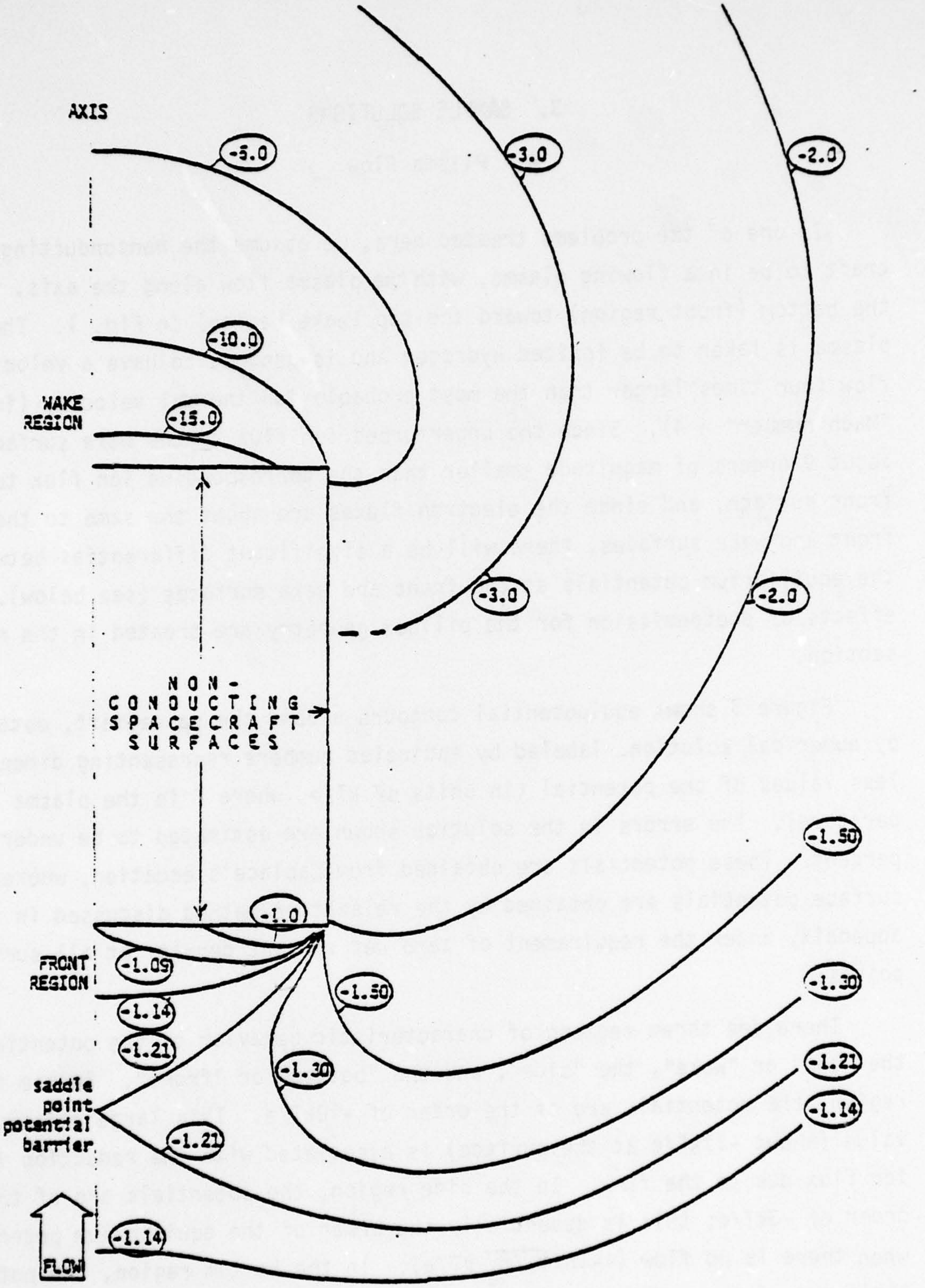

FIG. 3. DIFFERENTIAL CHARGING OF NONCONDUCTING SPACECRAFT BY PLASMA FLOW (EQUIPOTENTIAL CONTOURS IN UNITS OF kT/e)

points are thus not equipotential. Note that there is a saddle-point in the front region, that is, a potential barrier for electrons. The barrier height is between 10 percent and 20 percent higher than the potentials at the nearest (front) surface points. This feature is caused by the interaction between the relatively-large-magnitude wake potentials and the relatively-low-magnitude front potentials. The dashed part of the contour labeled "-3.0" near the side surface indicates that there is more complicated fine structure (variations of potential along the side surface) than is shown in the figure. The potentials along the top surface fall off to the right. The potentials along the bottom surface first fall with radius, then rise sharply as the corner is approached. This may be a "corner effect."

On the basis of these results, one would expect a relatively small body in the ionosphere, such as a thin antenna or boom painted with nonconducting paint, or a painted or insulated object in the very near wake of a spacecraft (or the spacecraft surface itself if it is a dielectric) to become highly negatively charged, to potentials of the order of volts in the ionosphere.

In the solar wind, the above calculation could apply to an entire spacecraft since the Debye length is large. However, the ion Mach numbers are of the order of 10 rather than 4, which would lead to negative dimensionless wake potentials an order of magnitude larger than the wake potentials shown in Fig. 3 (It is shown in the appendix that the potential difference in units of kT/e generated by the flow should be of the order of the square of the ion Mach number of the flow.) This means that for T=10ev in the solar wind, one may have kilovolt potential differences between the wake and front surfaces. The electric fields due to this differential charging may significantly disturb measurements of low-energy plasma electrons, for example on the HELIOS spacecraft (<u>Rosenbauer, 1973</u>).

B. Photoemission

In the second problem treated here, the bottom surface of the pillbox is assumed to be illuminated by the sun (along the axial direction) and to emit photoelectrons, while the sides and top of the pillbox are dark and nonemitting. (The axial direction of illumination is appropriate for maintaining axial symmetry. In the solar wind the plasma flow and solar illumination are in the same direction.) The ambient plasma contributions are not considered simultaneously with the photoemission. The photoelectrons are assumed to be emitted isotropically and monoenergetically, with 1 ev of kinetic energy. All points and their associated areas on the bottom surface are emitting except for the corner point, for instance, Nos. 10, 11, and 12 in Fig. 2. In the actual problem the "terminator" was put at $R=0.95R_o$. (The terminator is not put exactly at the corner $R=R_o$, for numerical reasons.) The time-dependent method is used, together with the outside-in method discussed in the appendix. At zero time, there is no charge on any surface. As time increases from zero, emitted photoelectrons from the surface at first escape to infinity, leaving behind positive charge and causing the bottom surface to acquire a positive potential. As this potential builds up to a value of the order of a volt (the ejection energy), the photoelectrons no longer all escape to infinity, but begin to return to the spacecraft. (It is of interest that the potential is found to rise significantly above one volt - see below.)

Figure 4 shows the time-behavior of the potential of the center point on the bottom surface (for instance, No. 12 in Fig. 2). The potential in volts is plotted against dimensionless time, where the scale time t_o in seconds may be written as $1.11\ V_o/(R_oJ_o)$, where V_o is the kinetic energy of emission in volts, R_o is the scale dimension of the emitting area in cm, and J_o is the emitted current density in picoamp per cm^2. Thus, for a spacecraft radius $R_o=50$ cm, photoemission current density $J_o=1000$ picoamp/cm^2, and emission energy $V_o=1$ volt, the scale time is $t_o=2.22 \times 10^{-5}$ sec. (The ordinate in Fig. 4 scales with the ejection energy.) The potential along the bottom surface is not uniform, but is maximum in the center and falls off with radius (see Fig. 5, which indicates an approximate drop of

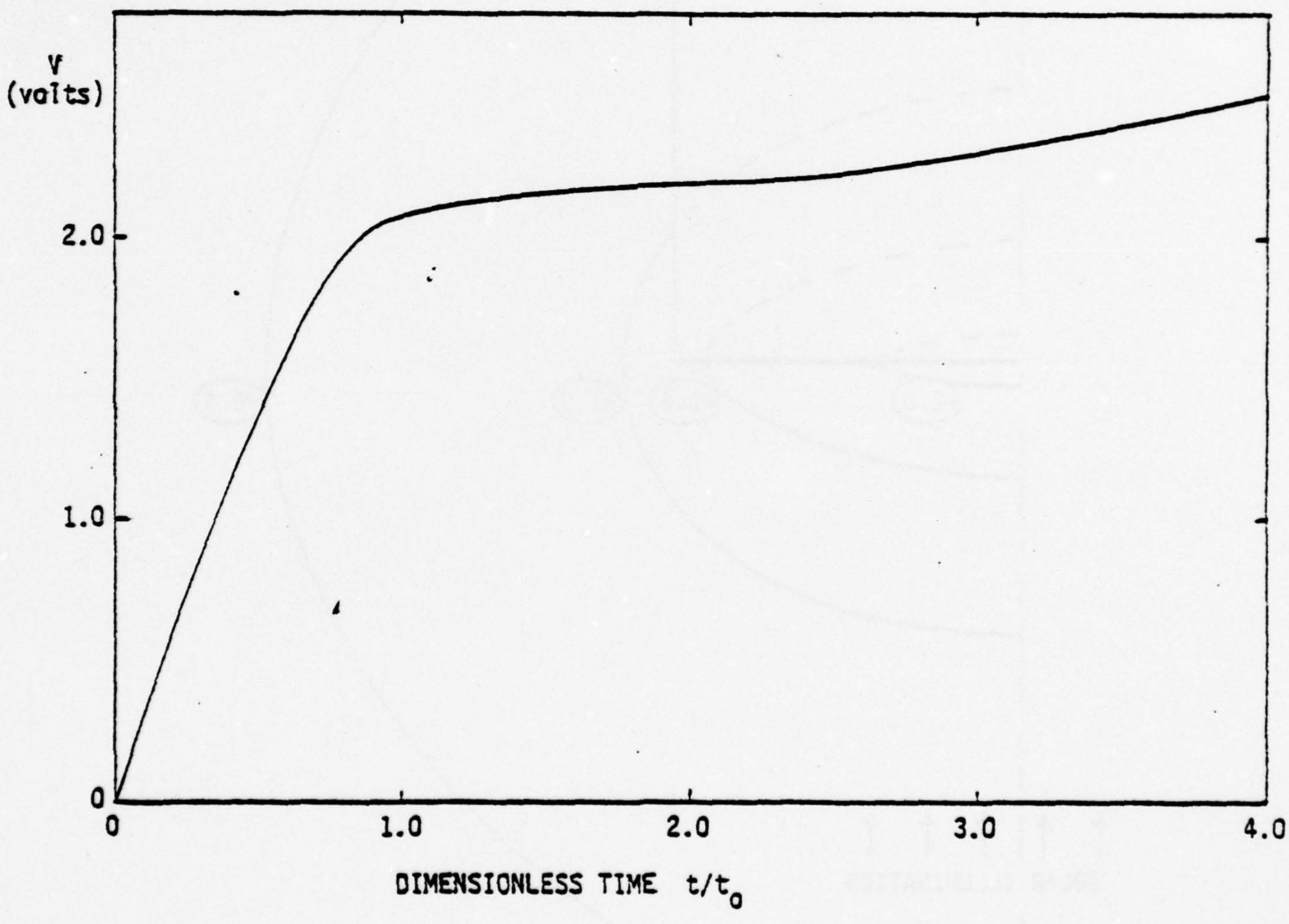

FIG. 4. POTENTIAL OF CENTER POINT VERSUS DIMENSIONLESS TIME

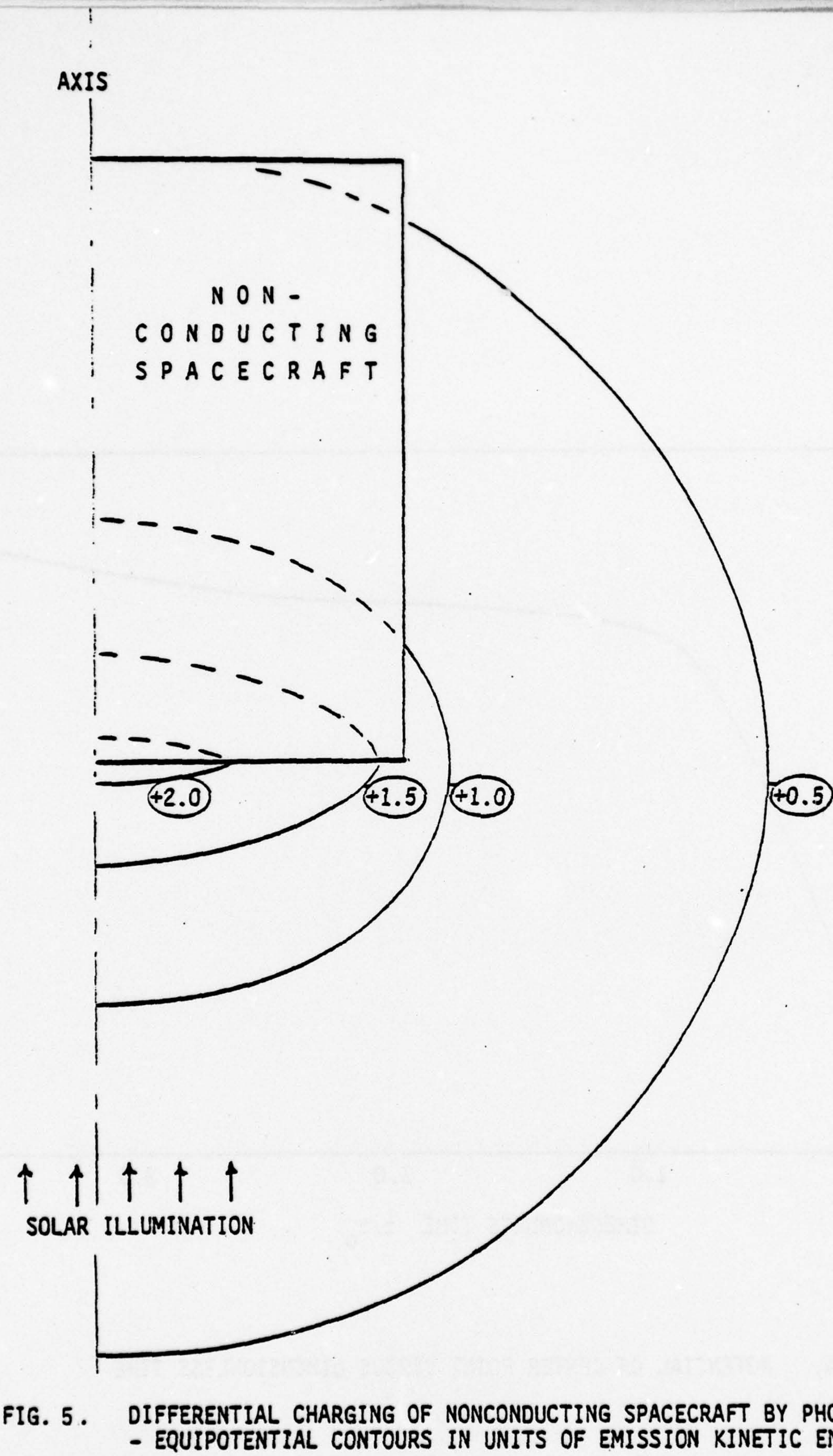

FIG. 5. DIFFERENTIAL CHARGING OF NONCONDUCTING SPACECRAFT BY PHOTOEMISSION - EQUIPOTENTIAL CONTOURS IN UNITS OF EMISSION KINETIC ENERGY - time=$2t_0$

30 percent over the illuminated area). The central potential rises to an approximate plateau of slightly over 2 volts in about one characteristic time t_o, and, after an interval of approximately constant behavior between t_o and $3t_o$, continues to increase but much more slowly than the initial rate.

It is a curious fact that the potential should rise to more than 2 times the ejected energy potential-equivalent, rather than to exactly the ejected energy as may be expected purely on the basis of conservation of energy without consideration of the nature of the trajectories, surface, and potential distribution. The computer results show a transfer of electrons from the illuminated areas to the small dark area associated with the corner point; that is, emitted electrons are pulled back, but cross the local terminator at $R=0.95R_o$ and hit the dark corner area rather than return to other points on the emitting area. The charge on this corner area continues to increase negatively, while the illuminated-surface charge increases positively. No charges are deposited on the top and side dark surfaces. A similar build-up to more than the ejection energy has been observed in computer experiments performed by De and Criswell (1977) and by Pelizzari and Criswell (1977) in studies of the photoelectric charging of locally-sunlit areas in the dark lunar terminator region. A possible explanation of this "excess" charging phenomenon is proposed here as follows.

The effect is appropriate to the problem of electron emission from a restricted area of a nonconducting surface, with no compensating electron flux from an external plasma. It also depends on the returning electrons "sticking" where they hit. After the initial potential buildup, despite the deposition of negative charge on the dark side of the terminator, the surface potential falls off monotonically from the central value but remains positive as one goes into the dark region. That is, the surface gradient (tangential electric field) remains finite and continuous across the terminator. (If the emitting area were a conductor, the surface gradient would be discontinuous and singular across the terminator.) Thus, there is a finite interval straddling the terminator, such that within this interval electrons can cross the terminator from a sunlit point (moving "uphill") to a dark point where they are held fast, without using up their kinetic energy.

A finite rate of transfer across the terminator from the sunlit area to the dark area can thus occur as long as the surface potential gradient is finite at the position of the terminator, regardless of how high the central potential of the sunlit area becomes. The transfer rate should approach zero as the gradient at the terminator approaches infinity. Whether this process is self-limiting, that is, whether the gradient at the terminator becomes infinite within a finite time, is presently unknown. The key point is probably that the sunlit area cannot be strictly an equipotential surface. It may be approximately so over most of its area, due to electron transport tending to maintain equipotentiality (De and Criswell, 1977; Pelizzari and Criswell, 1977), but the potential should fall off in the vicinity of the terminator.

Figure 5 shows a few equipotential contours around the spacecraft. (Only half of the spacecraft is shown, since it is symmetric about the axis.) The potential contours are taken from the solution of the foregoing problem, at the time $t=2t_0$. At this time the bottom-surface potential is approximately at its plateau value (Fig. 4) of about 2 volts, and is undergoing its smallest rate of change. The source of the field is a nonuniform disk of charge at the bottom surface, with positive charge on the left side of the terminator, and negative charge on the right. The equipotentials are symmetric about the plane of the disk since the spacecraft dielectric constant was assumed to be equal to the free-space value (no polarization effects).

APPENDIX A.

COMPUTER APPROACHES

The computer program listed in Appendix B was developed for the study of the sheath about a pillbox-shaped spacecraft. The sources of the sheath electric field include spacecraft surface potentials and space charge due to the ambient plasma. The program has options to include the effects of insulating as well as conducting spacecraft surfaces, and of electron emission due to photoelectric or (with minor modifications) secondary processes. The program can solve a coupled Poisson-Vlasov system of nonlinear partial-differential-integral equations, and uses a special iteration algorithm to obtain self-consistency between the Poisson and Vlasov solutions. This yields distributions of electric potential and ion and electron density. The "inside-out" method, which follows ion and electron trajectories backward to their origin at the body surface or in the undisturbed plasma, may be used as one option to compute the necessary moment-integrals for particle density and flux at arbitrarily chosen points. This is useful for contributions from the plasma. An "outside-in" approach is a useful alternative option for contributions from surface emission. A grid is used to define the spatial distributions of potential and density in the space about the spacecraft. The approach is applicable to a larger range of the parameters than other available approaches.

Figure A1 illustrates the nature of the grid representation used for a satellite having the form of a finite-length cylinder. The geometry is axially-symmetric, with the axis shown as the vertical dotted boundary line on the left. The boundary condition representing the condition at infinity is applied to the other boundary lines of the grid. There is an inner boundary representing the satellite surface, on the grid points of which the surface potentials (ϕ_a, ϕ_b, etc.) are defined. The grid points are at the intersections of the grid lines at constant r and constant z. Associated with each grid point is a volume of revolution in the shape of a torus of rectangular cross-section (shown as shaded boxes surrounding grid points).

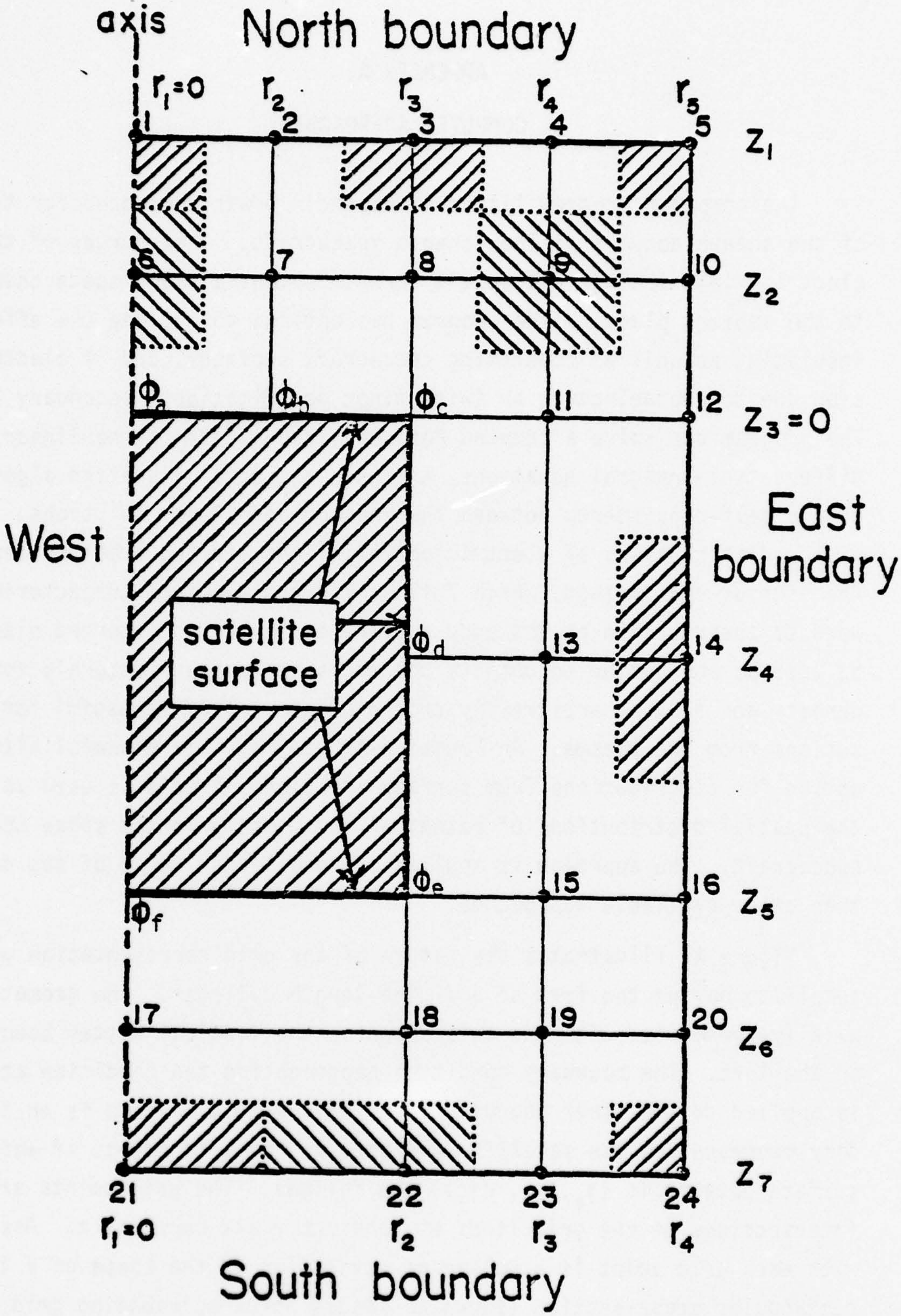

FIG. A1. GRID REPRESENTATION

The particle fluxes calculated at the satellite-surface grid points determine the equilibrium surface-potential distribution (ϕ_a, ϕ_b, etc.), with current density balance at individual grid points in the case of insulating sections, and current balance over the conducting areas in the case of conducting sections. The flux calculation is based on the following.

Assuming a fixed electric field configuration (potential-values given at all grid points), the flux may be written as a triple integral over velocity space, of the form

$$j(\vec{r}) = \int\int\int f(\vec{r},\vec{v})\ v_n\ d^3\vec{v} \tag{A-1}$$

where $\vec{v}$ is the vector velocity of a particle passing through the point $\vec{r}$, and where v_n is the component of $\vec{v}$ normal to the surface at $\vec{r}$. The distribution function $f(\vec{r},\vec{v})$ is the density of points in six-dimensional phase space. In the absence of collisions and time-variations, f depends only on the constants of motion and is constant along any orbit.

The region of interest may be considered to be enclosed by a composite source surface on all points of which f is assumed to be known. There is an external boundary surface at infinity where f has the unperturbed ambient value $f_\infty(\vec{v}_\infty)$, and internal source surfaces on the spacecraft where f has the value $f_s(\vec{r}_s,\vec{v}_s)$ in accord with the emission from these surfaces. Separate contributions to the flux j from infinity and from the spacecraft surfaces can be written as independent integrals of the form of Eq. (A-1), where each is comprised only of contributions from its associated source surface. In order to determine whether a specified velocity $\vec{v}$ (at $\vec{r}$) connects with infinity or with a point elsewhere on the spacecraft surface, the orbit is followed backward in time to its source (all orbits being dynamically reversible). This is generally an appropriate task for a computer and is the heart of the "inside-out" method.

Since symmetric velocity distributions are of interest the polar form of velocity space is used in Eq. (A-1). Thus, the flux at a surface point $\vec{r}$ may be written as

$$j(\vec{r}) = \iiint_{\text{hemisphere}} f\, v^3\, dv \cos\alpha\, d\Omega \tag{A-2}$$

where the velocity-space volume element has been expressed in terms of a local velocity-coordinate system by

$$d^3\vec{v} = v^2\, dv\, d\Omega, \qquad d\Omega = \sin\alpha\, d\alpha\, d\beta \tag{A-3}$$

Here, v, α, and β denote the magnitude, polar angle, and azimuthal angle, respectively, of the velocity-vector $\vec{v}$, where the definitions of the angle variables α and β are illustrated in Fig. A2. The flux j is the component of the flux vector $\vec{j}$ in the direction of the chosen axis, e.g., the normal to the surface at the point $\vec{r}$. In Eq. (A-2) the subscript "hemisphere" denotes that the angular integration is over the hemisphere of outgoing directions (2π steradians).

Contributions from Infinity

Assuming at infinity a Maxwellian-with-drift velocity distribution, with temperature T, particle mass m, particle density n_o, and drift Mach number M, the flux contribution from ambient particles may be written:

$$j_\infty = n_o \left(\frac{kT}{2\pi m}\right)^{1/2} \int_{\mathrm{Max}(\phi,o)}^{\infty} dE\, (E-\phi) \iint e^{-U}\, \frac{\delta \cos\alpha\, d\Omega}{\pi} \tag{A-4}$$

where

$$U = E + M^2 - 2ME^{1/2} \cos\theta_\infty \tag{A-5}$$

Here, δ is a "cut-off" function which is unity if the orbit connects with infinity, and is zero otherwise; E and ϕ denote local dimensionless total energy and electric potential, normalized by kT and kT/e, respectively; θ_∞ is the angle between $\vec{v}_\infty$ and the drift direction at infinity, where $\vec{v}_\infty$ is the velocity at infinity of the orbit characterized locally by E, α, and β.

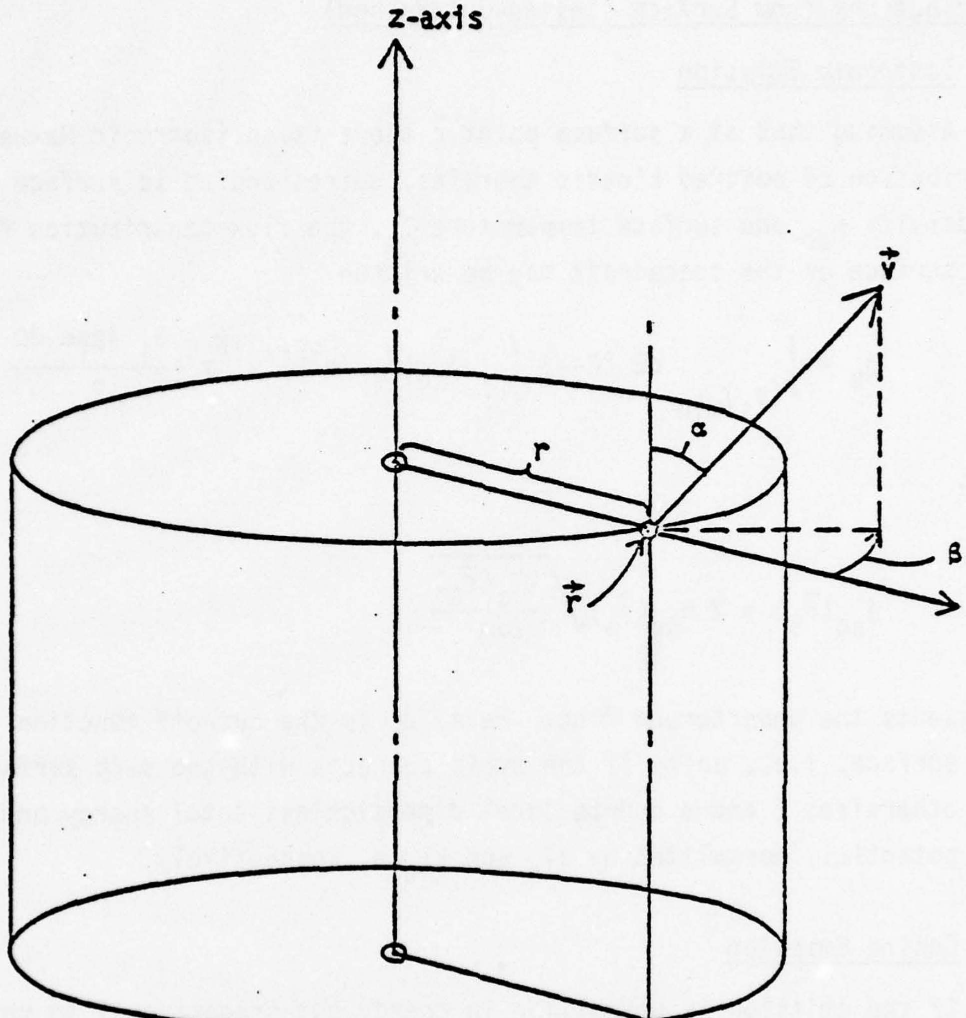

FIG. A2. ANGLE VARIABLES IN VELOCITY SPACE

Contributions from Surface (Inside-Out Method)

A. Isotropic Emission

Assuming that at a surface point $\vec{r}$ there is an isotropic Maxwellian distribution of emitted kinetic energies, corresponding to surface particle density n_{so} and surface temperature T_s, the flux contribution from the s-th surface on the spacecraft may be written

$$j_s = \int_{(\phi_s)_{min}}^{\infty} dE\ (E-\phi) \int\int j_{so}(\vec{r}_s)\ e^{-E+\phi_s(\vec{r}_s)}\ \frac{\delta_s \cos\alpha\ d\Omega}{\pi} \qquad \text{(A-6)}$$

where

$$j_{so}(\vec{r}_s) = 2\ n_{so}(\vec{r}_s)\sqrt{\frac{kT_s(\vec{r}_s)}{2\pi m}} \qquad \text{(A-7)}$$

represents the unperturbed flux. Here, δ_s is the cut-off function for the s-th surface, i.e., unity if the orbit connects with the s-th surface, and zero otherwise; E and ϕ denote local dimensionless total energy and electric potential, normalized by kT_s and kT_s/e, respectively.

B. Cosine Emission

If the emission is Maxwellian in energy but proportional to the cosine of the angle made with the surface normal rather than isotropic, then Eq. (A-6) is replaced by

$$j_s = \int_{(\phi_s)_{min}}^{\infty} dE\ (E-\phi) \int\int \frac{4}{3}\, j_{so}(\vec{r}_s)\ e^{-E+\phi_s(\vec{r}_s)}\ \delta_s \frac{3\cos^2\alpha\ d\Omega}{2\pi} \qquad \text{(A-8)}$$

where the function j_{so} is still given by Eq. (A-7), but where $(4/3)j_{so}$ represents the new unperturbed flux.

Contributions from Surface (Outside-In Method)

One can also use an "outside-in" approach, i.e., following trajectories forward in time, to compute fluxes or densities. This approach is of course equivalent to and would provide the same results as the inside-out method, but may be computationally more efficient in certain problems. In particular, it should be more efficient for calculating flux contributions from the surface. For this approach, let dA represent an element of surface receiving contributions from another element of surface dA'. Moreover, let $dF_{AA'}$ denote the fraction of emitted trajectories from dA' which reach dA. Morover, let j_A denote the flux received at A, while $j_{A'}$ denotes the flux emitted at A'. Then by conservation of particles we may write

$$dj_A \cdot dA = dF_{AA'} \cdot j_{A'} \cdot dA' \tag{A-9}$$

Thus, the flux received at point A due to emission from all emitting points A' is given by the integral

$$j_A = \int_{\text{all } A'} j_{A'} \cdot \frac{dF_{AA'}}{dA} \cdot dA' \tag{A-10}$$

where dF/dA is determined by trajectory calculations. The foregoing integrals may be implemented numerically as shown below.

Assuming that the ambient plasma flows along the z-direction with ion Mach number M, the dimensionless velocity-distribution function at infinity may be written:

$$f_o = \frac{1}{\pi^{3/2}} \exp(-U) \tag{A-11}$$

where (repeating Eq. (A-5))

$$U \equiv E + M^2 - 2ME^{1/2} \cos\theta_\infty \tag{A-5}$$

The flux integral, namely,

$$j(\vec{r}) = \int\int\int f(\vec{r},\vec{v})\, v_n d^3\vec{v} \quad , \qquad \text{(A-1)}$$

where f is the distribution function and v_n is the component of the particle velocity normal to the surface, may be approximated for numerical evaluation by the inside-out method using a quadrature sum as follows:

$$j \cong \sum_i^I \sum_j^J \sum_k^K A_{ijk}\, \delta_{ijk} \frac{(E_k - \phi)}{\pi^{3/2}} \exp\,(-U_{ijk}) \qquad \text{(A-12)}$$

where U is defined by Eq. (A-5), ϕ is the dimensionless local potential, and the 3 indices refer to discrete values of the 3 components of velocity. The discrete values are chosen in accord with a Gaussian quadrature scheme, and the coefficients A_{ijk} are proportional to the associated weights and other factors.

The flux integral for the outside-in method, namely, Eq. (A-10), may be evaluated by a four-fold quadrature sum as follows:

$$j \cong \sum_i^I \sum_j^J \sum_k^K \sum_a B_{ijka} \cdot j_a \cdot \frac{F_a}{A} \cdot A_a' \qquad \text{(A-13)}$$

Here, in addition to the sum over the 3 velocity indices, we indicate a summation over contributing finite areas A_a'. The area A denotes the finite area at the point of interest; F_a denotes the fraction of emitted trajectories (assumed all emitted from the center of A_a') which intersect A; j_a denotes the emitted flux at A_a'; and B_{ijka} is the appropriate coefficient.

For the purpose of computing surface-potential distributions over a spacecraft surface with a variation of surface properties, we associate

separate differential area segments with the surface grid points defining the surface, as illustrated in Fig. 2. The areas associated with the surface grid points in Fig. 2 are labeled A_1, A_2, etc. These may represent areas with different conductivities and different emission properties, either intrinsic or depending on their geometric position and orientation (as in photoemission).

For a given distribution of surface properties, the equilibrium potential distribution may be determined by relaxation as follows. A distribution of surface potentials (at the grid points) is initially assumed. This leads by trajectory computations to a distribution of fluxes of ions and electrons at the surfaces. The ion and electron fluxes can be determined numerically using summations represented by Eq. (A-12) or (A-13). On each conducting area (or collection of connected conducting areas) the potential must be adjusted so that the net current to the area is zero. At nonconducting points the net current density must be zero. These adjustments can be accomplished by an iterative relaxation procedure. Thus, the initial guesses for the surface potentials are modified according to the sign and magnitude of the net current or current density. The essence of the relaxation procedure, for a totally nonconducting spacecraft, is the iterative algorithm:

$$\phi_k^{N+1} = \phi_k^N + \alpha \cdot CD_k^N \tag{A-14}$$

where the superscript N denotes the N-th iteration, and where ϕ_k is the dimensionless potential at the k-th surface point, CD_k is the net current density at that point, and α is the relaxation parameter.

For the time-dependent approach, the charge associated with the finite area A is given by the time-integral:

$$Q_A = \int j \cdot A \cdot dt \tag{A-15}$$

This charge is used as the source term at the center of the associated volume of revolution indicated in Fig. A1. In a step-by-step procedure, during one half of each cycle, the fluxes are evaluated in accord with Eq. (A-12)

or (A-13); in the other half of the cycle the charges are updated in accord with the discretized form of Eq. (A-15).

Charging of the Wake Surface in Plasma Flow

One may estimate the potential acquired by the wake surface of a non-conducting spacecraft in a plasma flow, as follows. Assume that the dimensionless potential ϕ is negative, and that the electron and ion fluxes to the wake surface are given, respectively, by

$$j_e \cong n_o \sqrt{kT/2\pi m_e}\, \exp(+\phi) \qquad \text{(A-16)}$$

and

$$j_i \cong n_o \sqrt{kT/2\pi m_i}\, \{\exp(-M^2)/2M^2\} \qquad \text{(A-17)}$$

where M is the ion Mach number. For large values of M, the bracketed expression in Eq. (A-17) is the asymptotic form of $\{\exp(-M^2) - \sqrt{\pi}M \operatorname{erfc} M\}$, the exact factor for the neutral ion flux. Equating Eqs. (A-16) and (A-17), we may write:

$$-\phi \cong M^2 + \ln 2M^2 + \ln\sqrt{m_i/m_e} \qquad \text{(A-18)}$$

This will yield an overestimate, since the ion flux is actually larger than that given by Eq. (A-17). For M=4 and m_i/m_e=1836 for hydrogen, we obtain ϕ~-23, versus the self-consistent value ϕ~-17 presented in Section 3. For M=10, the potential is ϕ~-109. Hence, for large M we may obtain a good estimate by keeping only the first term on the right-hand side. Then the potential difference generated by the flow becomes independent of the temperature and may be estimated by $m_i v^2/2e$, where m_i is the ion mass and v is the velocity of the flow. Thus, the potential difference in volts may be estimated by $0.00519 \cdot m(au) v^2$(km/sec) where m(au) is the ion mass in atomic units and v(km/sec) is the flow velocity in km/sec. Thus, in the ionosphere, assuming m(au)=16 (O^+ ions) and v=7 km/sec (orbital velocity), the potential difference is about 4.0 volts. In the solar wind, with m(au)=1 (protons) and v=440 km/sec, the potential difference becomes 1.0 kilovolt.

APPENDIX B.

COMPUTER PROGRAM

```
      PROGRAM PARKO2(INPUT,OUTPUT,TAPE60=INPUT,TAPE61=OUTPUT,PUNCH)
C
C     LEE W. PARKER, INC.
C     252 LEXINGTON ROAD, CONCORD, MASSACHUSETTS 01742
C
C
C     SATELLITE DIFFERENTIAL CHARGING PROBLEM
C     ITERATION ON SURFACE POTENTIALS TO ACHIEVE ZERO NET CURRENT
C        DENSITY AT EACH SURFACE POINT
C        PILLBOX OR THICK DISK (-ZFRONT = THICKNESS )
C
      COMMON JJN,IIN,JJS,IIS,NIOT,RNT(50),ZNT(50),RST(50),ZST(50),
     1 RZ( 500,2),PHTN(20,20),PHTS(20,20),CD( 500),IFIRST,M
      COMMON/F_D/NCOLSN,NCOLSE,NCOLSS,NROWSN,X(500),NROWSS,DEBYE,DEBYE2,
     1 RHON1(50),RHOE(50),RHOS1(50),ZN(50),    ZS(50),PHI(50),NGAP,NDISK
      COMMON/DEN/NPRINT,MD,MC,MA,MB,ME,STEP,RSAVE,ZSAVE,ALPHA,BETA,EE,
     1 XMSAVE,RADIUS,FLUXT(3,20),FLUXE(3,20),FLUXEM(3,20),CDSAVE(100)
      COMMON/DEN1/IT,MAME,NEWPHI,ISAVE,NGR,NGROUP( 500),DSAVE( 500)
     1 , ZFRONT,NSS,NROWS,NPHOTO,AREA(100),EFF_UX(100)
      COMMON/PZRZ/PA(50),PB(50),ZA(50),ZB(50),IIA,JJA,IIB,JJB
      DIMENSION DATE(20),X1( 500),VPROBE(20),CAPINV(30,30),CHARGE(30)
      DIMENSION SIGMA(30)
C
C FOR DENSITY AND CURRENT CALCULATIONS
C INTEGER INPUTS ARE NPRINT,MD,MC,MA,MB,ME.  NPRINT=0,1,2,3 FOR PRINTING
C NO INTERMEDIATE DATA, ESCAPING INDICES, FIRST AND LAST STEPS, AND ALL
C STEPS.  MD=0,1 FOR ONE SPACE POINT (RSAVE,ZSAVE) OR SEVERAL SPACE
C POINTS (READ IN R,Z VALUES).  MC=0,1 FOR CHARGE DENSITY OR CURRENT DE
C SITY. MA=0 MEANS ONE TRAJECTORY (READ RSAVE,ZSAVE,ALPHA,BETA,EE).  ME
C MEANS ONE ENERGY (READ EE).  OTHERWISE, MA,MB,ME, DENOTE THE NUMBER O
C ALPHA-INTERVALS, BETA-INTERVALS, AND ENERGY-INTERVALS.
C
      L=60
      M=61
      IPUNCH=0
      IPUNCH=1
      ISAVE=0
      DEBYE2=0.
      STEP=.1
      PI=3.14159265358979
    1 READ(L,9998) DATE
 9998 FORMAT(20A4)
      IF(EOF(L)) 99,15
   15 WRITE(M,9999) DATE
 9999 FORMAT(1H1,20A4)
      READ(L,111)   NCOLSN,NCOLSE,NCOLSS,NROWSN,NROWSS,NROWS,NPHOTO
      NROWSE=1
      JJN=NCOLSN+NCOLSE
      IIN=NROWSN+1
      JJS=NCOLSS+NCOLSE
      IIS=NROWSS+1
```

1


```
      NDISK=NCOLSN+NCOLSS
      NGAP=NDISK+1
      NSS=NCOLSN
      IF(NROWS.GT.0) NSS=NCOLSN+NCOLSS+NROWS+1
      NTOT=IJN*IIN+JJS*NROWSS
      READ(L,222)  (RHON1(J),J=1,NCOLSN)
      READ(L,222)  (RHOE(J),J=1,NCOLSE)
      READ(L,222)  (RHOS1(J),J=1,NCOLSS)
      READ(L,222)  (ZN(I),I=1,NROWSN)
      READ(L,222)  (ZS(I),I=1,NROWSS)
      RADIUS=RHON1(NCOLSN)
      ZFRONT=0.
      IF(NROWS.GT.0) ZFRONT=ZS(NROWS)
********** MODIFICATION FOR FINITE DISK THICKNESS
      READ(L,222) DEBYE,ALPH,RBOUND,ZNBOUN ,ZSBOUN ,RWAKE
      WRITE(M,220) DEBYE,ALPH,RBOUND,ZNBOUN ,ZSBOUN ,RWAKE,ZFRONT
      READ(L,111)  IIS,IT,NEWPHI,MAME,ITALL
      WRITE(M,110) IIS,IT,NEWPHI,MAME,ITALL
      ITMAX=IIS+IT
      IF(NPHOTO.GT.0) WRITE (M,227) NPHOTO
      IF(IT.EQ.0) TIME=0.
      IRELAX=0
C     SET IRELAX=1   FOR RELAXATION METHOD
C     SET IRELAX=0   FOR TIME-DEPENDENT METHOD
      DO 17 N=1,NTOT
      CD(N)=0.
      X(N)=X1(N)=0.
      DSAVE(N)=0.
      IF (N.GT.NSS) GO TO 17
      PHI(N)=0.
      CHARGE(N)=0.
17    CONTINUE
      NTOTAL=NTOT
      IF(ISAVE.GT.0) READ(L,52)NTOT,NTOT,(DSAVE(N),N=1,NTOT)
      IF(IT.EQ.0) GO TO 21
      READ(L,52) IT,NTOTAL,(X(N),N=1,NTOTAL)
      IF(DEBYE.EQ.0.) READ (L,222) (PHI(J),J=1,NSS)
      IF (DEBYE.EQ.0.)READ (L,222) TIME
      IF (DEBYE.EQ.0.)READ (L,222) (CHARGE(J),J=1,NSS)
      IF(DEBYE.GT.0..AND.NTOTAL.NE.NTOT) GO TO 18
      GO TO 21
   18 WRITE(M,668) NTOT,NTOTAL
  668 FORMAT(////1X,41HTROUBLE - INCORRECT NUMBER OF POINTS READ,5X,2I5)
      GO TO 99
C
21    IF(NEWPHI.EQ.0.AND.DEBYE.GT.0.)
     1 WRITE(4,230) DEBYE,ALPH,ITMAX,IT,(N, X(N),N=1,NTOT)
      IF(NEWPHI.GT.0.AND.DEBYE.GT.0.)
     1 WRITE(4,2301)DEBYE,ALPH,ITMAX,IT,(N, X(N),N=1,NTOT)
C
C RESCALE GRID LINE POSITIONS TO FIT WITHIN INPUT BOUNDS.
```

2

```
C
      RATIO=(RBOUND-RADIUS)/(RHOE(NCOLSE)-RADIUS)
      DO 23 J=1,NCOLSE
      IF(RATIO.GT.0.)  RHOE(J)=RADIUS + RATIO*(RHOE(J)-RADIUS)
23    CONTINUE
      RATIO=ZNBOUN/ZN(1)
      DO 24 I=1,NROWSN
      IF(RATIO.GT.0.)    ZN(I)=RATIO*ZN(I)
24    CONTINUE
      RATIO=ZSBOUN/ZS(NROWSS)
      IF(NROWS.GT.0) RATIO=(ZSBOUN -ZFRONT)/(ZS(NROWSS)-ZFRONT)
      DO 25 I=1,NROWSS
      IF (RATIO.LE.0) GO TO 25
      IF (NROWS.EQ.0) ZS(I)=RATIO*ZS(I)
      IF (NROWS.GT.0) ZS(I)=ZFRONT+RATIO*(ZS(I)-ZFRONT)
25    CONTINUE
C
      IF (NROWS.EQ.0) PHI(NGAP)=.5*(PHI(NCOLSN)+PHI(NCOLSN+1))
      WRITE (M,112) NCOLSN,NCOLSE,NCOLSS,NROWSN,NROWSS,NROWS
      WRITE(M,223)  (J,RHON1(J),J=1,NCOLSN)
      WRITE(M,224)  (J,RHOE(J),J=1,NCOLSE)
      WRITE(M,225)  (J,RHOS1(J),J=1,NCOLSS)
      WRITE(M,226)  (I,ZN(I),I=1,NROWSN)
      WRITE(M,228)  (I,ZS(I),I=1,NROWSS)
      WRITE(M,229)  (J,PHI(J),J=1,NSS)
110   FORMAT(1X,30HNO. OF ITS, IT, NEWPHI, MAXF =, 4I5,5X,
     1 16HALL POINTS EVERY , I3,24H-TH ITERATION AFTER IT=2)
111   FORMAT(16I5)
112   FORMAT(//1X,I3,25H COLUMNS (R-VALUES) NORTH/
     1          1X,I3,25H COLUMNS (R-VALUES) EAST /
     2          1X,I3,25H COLUMNS (R-VALUES) SOUTH/
     3          1X,I3,22H ROWS (Z-VALUES) NORTH/
     4          1X,I3,22H ROWS (Z-VALUES) SOUTH /
     5          1X,I3,26H SIDE ROWS (Z-VALUES) EAST)
220   FORMAT(1X,6HDEBYE=,F10.5,5X,5HALPH=,F10.5,
     1 10X,7HRBOUND=, F7.2,5X,8HZNBOUND=, F7.2,5X,8HZSBOUND=, F7.2,
********** MODIFICATION FOR FINITE DISK THICKNESS
     2 5X, 6HRWAKE=, F7.2/1X,7HZFRONT=, F7.2)
 221 FORMAT (1P9E10.3)
 222 FORMAT(9F10.5)
 223 FORMAT(//15H R-VALUES NORTH/(I3,1PE15.4))
 224 FORMAT(//14H R-VALUES EAST/(I3,1PE15.4))
 225 FORMAT(//15H R-VALUES SOUTH/(I3,1PE15.4))
 226 FORMAT(//15H Z-VALUES NORTH/(I3,1PE15.4))
 228 FORMAT(//15H Z-VALUES SOUTH/(I3,1PE15.4))
 227 FORMAT (1X,26HEMISSION INCLUDED. NEMOTO=,I7)
 229 FORMAT(//23H POTENTIALS ON SURFACES/(I3,1PE15.4))
230   FORMAT(/26H   CONVENTIONAL ITERATION,,
     1 30H  POTENTIALS WITH DEBYE NUMBER, F10.5,5X,6HALPH =,
     2 F10.5,5X,6HITMAX=,I4,5X,3HIT=,I4/(1X,I3,1PE15.4))
235   FORMAT(//1X,6HTIME =,F10.3/8X,11HPOTENTIALS,,4X,8HCHARGES,,7X,
```



```
     1 6HAREAS,,5X,27HAND CHARGE/AREA OF SURFACES / (1X,I2,1P4E15.4))
2301  FORMAT(/26HDEXP-IN-POISSON ITERATION,,
     1 30H  POTENTIALS WITH DEBYE NUMBER, F10.5,5X,6HALPH =,
     2 F10.5,5X,6HITMAX=,I4,5X,3HIT=,I4/(1X,I3,1PE15.4))
5050  FORMAT(1H0,I4,16H ORDER POTENTIAL)
5060  FORMAT(1H0,I4,21H ORDER DENSITIES      )
5070  FORMAT(1H0,I4,22H ORDER CURRENT DENSITY)
  52  FORMAT(2I5,1F7E10.3/(  8E10.3))
C
      DO 2 J=1,NCOLSN
    2 RNT(J)=RHON1(J)
      DO 3 J=1,NCOLSE
      JPN=J+NCOLSN
    3 RNT(JPN)=RHOE(J)
      DO 4 I=1,NROWSN
    4 ZNT(I)=ZN(I)
      ZNT(IIN)=0.
      DO 5 J=1,NCOLSS
    5 RST(J)=RHOS1(J)
      DO 6 J=1,NCOLSE
      JPS=J+NCOLSS
    6 RST(JPS)=RHOE(J)
      ZST(1)=0.
      DO 7 I=1,NROWSS
      IPS=I+1
    7 ZST(IPS)=ZS(I)
C
      IIA=IIN+1
      JJA=JJN+1
      IIB=IIS+1
      JJB=JJS+1
C
      RA(1)=RNT(1)
      RA(JJA)=RNT(JJN)
      DO 65 J=2,JJN
65    RA(J)=.5*(RNT(J-1)+RNT(J))
C
      ZA(1)=ZNT(1)
      ZA(IIA)=ZNT(IIN)
      DO 66 I=2,IIN
66    ZA(I)=.5*(ZNT(I-1)+ZNT(I))
C
      RB(1)=RST(1)
      RB(JJB)=RST(JJS)
      DO 67 J=2,JJS
67    RB(J)=.5*(RST(J-1)+RST(J))
C
      ZB(1)=ZST(1)
      ZB(IIB)=ZST(IIS)
      DO 68 I=2,IIS
68    ZB(I)=.5*(ZST(I-1)+ZST(I))
```


BEST AVAILABLE COPY

```
C
      WRITE (M,231) (J,RA (J),J=1,JJA)
      WRITE (M,232) (J,RB (J),J=1,JJB)
      WRITE (M,233) (I,ZA (I),I=1,IIA)
      WRITE (M,234) (I,ZB (I),I=1,IIB)
  231 FORMAT (//1X,30HINTERSTITIAL   R-VALUES, NORTH/(I3,1PE15.4))
  232 FORMAT (//1X,30HINTERSTITIAL   R-VALUES, SOUTH/(I3,1PE15.4))
  233 FORMAT (//1X,30HINTERSTITIAL   Z-VALUES, NORTH/(I3,1PE15.4))
  234 FORMAT (//1X,30HINTERSTITIAL   Z-VALUES, SOUTH/(I3,1PE15.4))
C     OUTPUT    RHO  AND  Z    ARRAYS
      DO 71 I=1,IIN
      DO 71 J=1,JJN
      JRZ=(I-1)*JJN+J
      RZ(JRZ,1)=RNT(J)
   71 RZ(JRZ,2)=ZNT(I)
      DO 73 I=1,NROWSS
      DO 73 J=1,JJS
      JRZ=JRZ+1
      RZ(JRZ,1)=RST(J)
   73 RZ(JRZ,2)=ZS(I)
      NTOT=JRZ
      DO 74 J=1,NCOLSN
      JRZ=JRZ+1
      R1=RA(J)
      R2=RA(J+1)
      IF (J.EQ.NCOLSN) R2=RADIUS
      AREA(JRZ-NTOT)=PI*(R2**2-R1**2)
      RZ(JRZ,1)=RNT(J)
74    RZ(JRZ,2)=0.
      IF (NROWS.EQ.0) GO TO 77
      JRZ=JRZ+1
      RZ(JRZ,1)=RNT(NCOLSN)
      RZ(JRZ,2)=0.
      AREA(JRZ-NTOT)=2.*PI*RADIUS*(ZB(1)-ZB(2))
      DO 75 I=1,NROWS
      JRZ=JRZ+1
      Z2=ZB(I+1)
      Z1=ZB(I+2)
      IF(I.EQ.NROWS) Z1=ZFRONT
      AREA(JRZ-NTOT)=2.*PI*RADIUS*(Z2-Z1)
      RZ(JRZ,1)=RST(NCOLSS)
      RZ(JRZ,2)=ZS(I)
75    CONTINUE
77    CONTINUE
      DO 76 J=1,NCOLSS
      JREV=NCOLSS-J+1
      JRZ=JRZ+1
      R1=RB(JREV)
      R2=RB(JREV+1)
      IF(J.EQ.1) R2=RADIUS
      AREA(JRZ-NTOT)=PI*(R2**2-R1**2)
```


BEST AVAILABLE COPY


```
      RZ(JRZ,1)=RST(JREV)
      RZ(JRZ,2)=0.
      IF(NROWS.GT.0) RZ(JRZ,2)=ZS(NROWS)
76    CONTINUE
      JRZNT=JRZ-NTOT
      IF (JRZNT.NE.NSS) WRITE(M,668) JRZNT,NSS
      DO 79 N=1,NSS
      EFFLUX(N)=0.
      IF(NPHOTO.EQ.0.OR.PEBYE.GT.0.) GO TO 79
      IF (N.GE.NCOLSN) GO TO 79
      EFFLUX(N)=1.
      IF(IT.GT.0) GO TO 79
      IF (IRELAX.GT.0) GO TO 79
      IFIRST=1
      N1=JJN*(IIN-1)
      CD(N1+N)=1.
      CALL FIELD
      CD(N1+N)=0.
      IFIRST=0
      DO 81 NC=1,NCOLSN
81    CAPINV(NC,N)=X(N1+NC)
      WRITE(M,777) N,(NC,CAPINV(NC,N),NC=1,NCOLSN)
777   FORMAT(//1X,27H INVERSE CAPACITANCE MATRIX /1X,I2,8(1X,I3,1PE12.4)
     1 /(3X,8(1X,I3,1PE12.4)))
   79 CONTINUE
      DO 78 N=1,NTOT
      IF (IT.GT.0) GO TO 78
      X(N)=0.
78    CONTINUE
      WRITE (M,888)
888   FORMAT(1H1)
      DO 82 N=1,NCOLSN
82    WRITE(M,777) N,(NC,CAPINV(NC,N),NC=1,NCOLSN)
C
      NFPP=(NTOT/300) +1
      DO 85 IP=1,NFPP
      WRITE (M,80)
   80 FORMAT (1H1,6X,1HI,4X,6H  R(I),4X,4HZ(I)  )
      DO 85 I=1,60
      K1=I+300*(IP-1)
      K2=K1+60
      K3=K2+60
      K4=K3+60
      K5=K4+60
      IF(K5.LE.NTOT)WRITE(M,8)K1,RZ(K1,1),RZ(K1,2),K2,RZ(K2,1),RZ(K2,2),
     1 K3,RZ(K3,1),RZ(K3,2),K4,RZ(K4,1),RZ(K4,2),K5,RZ(K5,1),RZ(K5,2)
      IF (K5.LE.NTOT) GO TO 85
      IF(K4.LE.NTOT)WRITE(M,8)K1,RZ(K1,1),RZ(K1,2),K2,RZ(K2,1),RZ(K2,2),
     1 K3,RZ(K3,1),RZ(K3,2),K4,RZ(K4,1),RZ(K4,2)
      IF (K4.LE.NTOT) GO TO 85
      IF(K3.LE.NTOT)WRITE(M,8)K1,RZ(K1,1),RZ(K1,2),K2,RZ(K2,1),RZ(K2,2),
```

```
    1 K3,RZ(K3,1),RZ(K3,2)
      IF (K3.LE.NTOT) GO TO 85
      IF(K2.LE.NTOT)WRITE(M,8)K1,RZ(K1,1),RZ(K1,2),K2,RZ(K2,1),RZ(K2,2)
      IF (K2.LE.NTOT) GO TO 85
      IF(K1.LE.NTOT)WRITE(M,8)K1,RZ(K1,1),RZ(K1,2)
   85 CONTINUE
    8 FORMAT (5(I9,F10.3,F3.3))
      NGRPS=1
      IF(ITALL.LE.1) ITALL=1
      IF(ITALL.LE.1) GO TO 1201
C
C DEFINE GROUPS 1, 2, ETC., IN THE WAKE, IN ORDER OF AXIAL DISTANCE FROM
      ZGROUP=ZN(NROWSN)
      NGR=1
      DO 12 N=1,NTOT
      NREV=NTOT-N+1
      NGROUP(NREV)=0
      IF(RZ(NREV,2).LE.0..OR.NREV.LE.JJN) GO TO 12
      IF(RZ(NREV,2).NE.ZGROUP) GO TO 13
      IF(RZ(NREV,1).GT.RWAKE) GO TO 12
      NGROUP(NREV)=NGR
      GO TO 12
13    ZGROUP=RZ(NREV,2)
***** TEMPORARY JUMP TO FORCE NGR=1 FOR ALL WAKE PTS.
      JUMP=1
      IF(JUMP.EQ.1) GO TO 12
      NGR=NGR+1
12    CONTINUE
      NGRPS=NGR
C
1201  CONTINUE
      IF (DERYE.EQ.0.) GO TO 14
      READ(L,566)NPRINT1,MD1,MC1,MA1,MB1,ME1,STEP1,RSAVE1,ZSAVE1,
    1 ALPHA1,BETA1,FE1,XMSAVE1
  566 FORMAT(3I1,I2,2I5,F5.3,6F10.5)
14    CONTINUE
      READ(L,666)NPRINT2,MD2,MC2,MA2,MB2,ME2,STEP2,RSAVE2,ZSAVE2,
    1 ALPHA2,BETA2,FE2,XMSAVE2
C
      IFIRST=0
      READ (L,52) IVMAX
      IF (IVMAX.EQ.0) GO TO 9
C
C READ AND WRITE INPUT PROBE POTENTIALS
      READ(L,52) IVMAX,IVTTS,(VPROBE(IV),IV=1,IVMAX)
      WRITE(M,52)IVMAX,IVTTS,(VPROBE(IV),IV=1,IVMAX)
C
C DO POTENTIALS
C
    9 DO 600 IV=1,IVMAX
      IF(IVMAX.GT.0) PHI(1)=VPROBE(IV)
```



```
      IF(IVMAX.GT.0) PHI(2)=VPROBE(IV)
C
10    DO 30 NG=1,NGRPS
      NGR=NG
      IF(IFIRST.EQ.0.OR.IT.LE.2.OR.ITALL.LE.1.OR.MOD(IT,ITALL).EQ.0)
    1   NGR=0
      DO 16 N=1,NSS
      IF (DERYE.GT.0.) GO TO 16
      IF (IFIRST.EQ.0) CD(NTOT+N)=0.
      IF (IT.EQ.0) CHARGE(N)=0.
      IF(IRELAX.EQ.0) CHARGE(N)=CHARGE(N)+ALPH*AREA(N)*CD(NTOT+N)
      SIGMA(N)=CHARGE(N)/AREA(N)
      IF(IRELAX.EQ.0) GO TO 16
      IF (IFIRST.EQ.0) X1(N)=PHI(N)
      FAC=1.
      IF (IFIRST.GT.0) PHI(N)=PHI(N)+SIGN(FAC,CD(NTOT+N))
      X1(N)=ALPH*PHI(N)+(1.-ALPH)*X1(N)
      PHI(N)=X1(N)
16    CONTINUE
      DO 19 N=1,NCOLSN
      DO 19 NC=1,NCOLSN
      IF (DERYE.GT.0.) GO TO 19
      PHI(N)=PHI(N)+CHARGE(NC)*CAPINV(N,NC)
19    CONTINUE
      IF(IFIRST.GT.0.AND.IRELAX.EQ.0) TIME=TIME+ALPH
      PCORN1=.5*(PHI(NCOLSN)+PHI(NCOLSN+1))
      PCORN2=.5*(PHI(NCOLSN+NROWS+1)+PHI(NCOLSN+NROWS+2))
      PHI(NCOLSN)=PCORN1
      PHI(NCOLSN+1)=PCORN1
      PHI(NCOLSN+NROWS+1)=PCORN2
      PHI(NCOLSN+NROWS+2)=PCORN2
      JCORN=0
      WRITE (M,235) TIME,(J,PHI(J),CHARGE(J),AREA(J),SIGMA(J),J=1,NSS)
      PHI(NSS+1)=PCORN1
      PHI(NSS+2)=PCORN2
      CALL FIELD
      IF(NGR.LE.1) WRITE(M,5050) IT
      NGO=1
11    IF (NGO.EQ.1.OR.DERYE.EQ.0) GO TO 45
      DO 20 N=1,NTOTAL
      IF(IFIRST.EQ.0)  X1(N)= X(N)
      X1(N)=ALPH* X(N)+(1.-ALPH)* X1(N)
   20 X(N)= X1(N)
45    DO 50 I=1,IIN
      DO 50 J=1,JJN
      INDX=J+(I-1)*JJN
   50 PHIN(I,J)=X(INDX)
      INDX=NROWSN*JJN
      DO 55 J=1,JJS
      INDX=INDX+1
   55 PHIS(1,J)=X(INDX)
```


BEST AVAILABLE COPY

```
      DO 55 I=2,IIS
      DO 56 J=1,JJS
      INDX=INDX+1
      PHIS(I,J)=X(INDX)
   56 CONTINUE
      IF(NGO.EQ.2) GO TO 500
      NGO=2
      WRTIE(M,120) IT,NG2
  120 FORMAT(///,1X,23HPOTENTIAL ARRAY - NORTH,5X,4HIT =,I3,3X,5HNGR =,
     1 I3)
      WRITE(M,2004) (RNT(J),J=1,JJN)
      DO 91 I=1,IIN
      WRITE (M,123)  I,ZNT(I),(PHIN(I,J),J=1,JJN)
   91 CONTINUE
      WRITE (M,122)
  122 FORMAT (///,1X,35H POTENTIAL ARRAY - SOUTH.            //)
      WRITE(M,2004) (RST(J),J=1,JJS)
 2004 FORMAT(/1X,2HR=,16F8.4/(/3X,16F8.4))
      DO 93 I=1,IIS
      WRITE (M,123)  I,ZST(I),(PHIS(I,J),J=1,JJS)
   93 CONTINUE
  123 FORMAT(/5H LINE,I4,5X,2HZ=,F8.4/(/7F16.8))
C
      GO TO 11
  500 CONTINUE
      IF(IPUNCH.GT.0)                 PUNCH 52, IT,NTOTAL,(X(N),N=1,NTOTAL)
      IF (IPUNCH.GT.0.AND.DEBYE.EQ.0.) PUNCH 52,IT,NSS
      IF (IPUNCH.GT.0.AND.DEBYE.EQ.0.) PUNCH 221,(PHI(N),N=1,NSS)
      IF (IPUNCH.GT.0.AND.DEBYE.EQ.0.) PUNCH 221;TIME
      IF (IPUNCH.GT.0.AND.DEBYE.EQ.0.) PUNCH 221,(CHARGE(N),N=1,NSS)
      IF(IT.GT.ITMAX) GO TO 1
      IF (DEBYE.EQ.0.) GO TO 31
C
C FIRST DO DENSITIES
C
      NPRINT=NPRINT1
      MD=MD1
      MC=MC1
      MA=MA1
      MB=MB1
      ME=MF1
      STEP=STEP1
C STOP IN DENSTY IF STEP LE ZERO.
      RSAVE=RSAVE1
      ZSAVE=ZSAVE1
      ALPHA=ALPHA1
      BETA=BETA1
      EE=EE1
      XMSAVE=XMSAVE1
      CALL DENSTY
      WRITE(M,664) NPRINT,MD,MC,MA,MB,ME,STEP,RSAVE,ZSAVE,ALPHA,BETA,EE,
```



```
     1 XMSAVE
  664 FORMAT(1X,22HNPRINT,MD,MC,MA,MB,ME=,6I4/
     1 1X,37HSTEP,RSAVE,ZSAVE,ALPHA,BETA,EE,XMACH=,7F10.5)
      WRITE(M,5060) IT
      IF (IFIRST.EQ.0.OR.NGR.EQ.0) GO TO 31
   30 CONTINUE
   31 CONTINUE
C
C THEN DO CURRENTS
C
      NPRINT=NPRINT2
****
      IF (IFIRST.GT.0) NPRINT=0
****
      MD=MD2
      MC=MC2
      MA=MA2
      MB=MB2
      ME=ME2
      STEP=STEP2
C STOP IN DENSTY IF STEP LE ZERO.
      RSAVE=RSAVE2
      ZSAVE=ZSAVE2
      ALPHA=ALPHA2
      BETA=BETA2
      EE=EE2
      XMSAVE=XMSAVE2
      CALL DENSTY
      WRITE(M,664) NPRINT,MD,MC,MA,MB,ME,STEP,RSAVE,ZSAVE,ALPHA,BETA,EE,
     1 XMSAVE
      IF(MC2.GT.0) WRITE(M,5070) IT
      IFIRST=IFIRST+1
      IT=IT+1
C
      IF(IT.GT.0.AND.IVMAX.GT.0.AND.MOD(IT,IVITS).EQ.0)   GO TO 600
      GO TO 10
  600 CONTINUE
C
   99 STOP
      END
```


BEST AVAILABLE COPY

```
      SUBROUTINE ROOT(A,B,X)
      COMMON JJN,IIN,JJS,IIS,NTOT,RNT(50),ZNI(50),RST(50),ZST(50),
     1 RZ( 500,2),PHIN(20,20),PHIS(20,20),CD( 500),IFIRST,M
C FIND ROOT OF  X+B*EXP(X)=A.  BY NEWTON METHOD.
      KPPA=0
      KPRB=0
      EPS=1.E-6
      XOLD=X=0.
      KMAX=1000
      DO 100 K=1,KMAX
      XOLD=X
      KPR=K
      F=X + B*EXP(X) - A
      FF=1. + B*EXP(X)
      DX=0.
      IF(FP.GT.0.) DX=-F/FP
      X=XOLD + DX
      DELTA=DX
      IF(ABS(X).GT.1.E-8) DELTA=DX/X
      IF(KPPA.GT.0) WRITE(M,1000) K,A,B,X,DX,DELTA,F,FP
1000  FORMAT(1X,22HK,A,B,X,DX,DELTA,F,FP=,I5,1P7E14.4)
      IF(ABS(DELTA).LT.EPS) GO TO 200
100   CONTINUE
      WRITE(M,3999) KMAX
3999  FORMAT(/////1X, 9HMORE THAN,I5,
     1 40H ITERATIONS IN ROOT. HENCE PROGRAM STOP.)
      STOP
C
200   CONTINUE
      PDELTA=100.*DELTA
      IF(KPRB.GT.0) WRITE(M,2000) EPS,X,PDELTA,KPR
2000  FORMAT(1X,35HCONVERGENCE IN ROOT WITHIN EPSILON=,1PE9.1,1H.,10X,
     1 3HX =,E12.4,7H WITHIN,E10.2,11H PERCENT IN,I4,12H ITERATIONS.)
      RETURN
      END
```



```
      SUBROUTINE FIELD
C
C     UNSYMMETRIC DISK FIELD
C      SEIDEL METHOD
C
C
C
      COMMON JJN,IIN,JJS,IIS,NTOT,RNT(50),ZNI(50),RST(50),ZST(50),
     1 RZ( 500,2),PHIN(20,20),PHIS(20,20),RHAND( 500),IFIRST,M
      COMMON/FLD/NCOLSN,NCOLSE,NCOLSS,NROWSN,X(500),NROWSS,DEBYE,DEBYE2,
     1 RHON1(50),RHOE(50),RHOS1(50),ZN(50),    ZS(50),PHI(50),NGAF,NDISK
      COMMON/A/CN,CS,CE,CW,C,V,  INDX,JSAT,RHO,Z,INDXN( 500),INDXS( 500)
     1,INDXE( 500),INDXW( 500),CONST( 500,6)
      COMMON/DEN1/IT,MAME,NEWPHI,ISAVE,NGR,NGROUE( 500),DSAVE( 500)
     1 , ZFRONT,NSS,NROWS,NPHOTO,AREA(100),EFF_UX(100)
C
C ASSUME ASYMPTOTIC MONOPOLE
      ALPHAF(RR,ZZ)=-ZZ/(ZZ**2 + RR**2)
      BETAF(RR,ZZ) =-RR/(ZZ**2 + RR**2)
C
      IF(IFIRST.EQ.0) GO TO 45
      IF(NEWPHI.EQ.0.AND.DEBYE.GT.0.) WRITE(M,222) DEBYE,IT,NGR,
     1 (N,RHAND(N),N=1,NTOT)
      IF(NEWPHI.GT.0.AND.DEBYE.GT.0.) WRITE(M,223) DEBYE,IT,NGR,
     1 (N,RHAND(N),N=1,NTOT)
222   FORMAT(1H1/18HOFIELD CALCULATION, 10X,
     1 41HINPUT=CHARGE DENSITIES WITH DEBYE NUMBER=,F10.5,
     2 10X,4HIT =,I3,3X,5HNGP =,I3/
     3 (28X,I3,1PE15.4))
223   FORMAT(1H1/18HOFIELD CALCULATION, 10X,
     1 41HINPUT=   ION DENSITIES WITH DEBYE NUMBER=,F10.5,
     2 10X,4HIT =,I3,3X,5HNGP =,I3/
     3 (28X,I3,1PE15.4))
C     NORTH + NORTHEAST REGION
C
C     FIRST POINT,  FIRST LINE
C
45    CONTINUE
      JSAT=0
      I=1
      IF(IFIRST.EQ.0) WRITE(M,333)
333   FORMAT(/////25H NORTH + NORTHEAST REGION///)
      IF(IFIRST.EQ.0) WRITE(M,334) I
334   FORMAT(//5H LINE,I3,93H                    N                    W
     1                 C                    E                    S)
      J=1
      INDX=J
      INDXN(INDX)=0
      INDXS(INDX)=J+IJN
      INDXE(INDX)= INDX + 1
      INDXW(INDX)=0
```


BEST AVAILABLE COPY

```
      Z=ZN(I)
      RHO=RNT(J)
      HN=0.
      HS= Z - ZN(I+1)
      HE=RNT(J+1)-RHO
      HW=0.
      ALPHA=ALPHAF(RHO,Z)
      CN=0.
      CS= 0.125*HE**2/HS
      CE= HS/4.
      CW=0.
      C= 0.125*HE*(HE/HS + 2.*HS/HE - ALPHA*HE)
      V= HS*HE**2/16.
      CALL PRINT
C
C      MIDDLE POINTS,  FIRST LINE
C
      JMAX=JJN-1
      DO 5 J=2,JMAX
      INDX=J
      INDXN(INDX)=0
      INDXS(INDX)=J+JJN
      INDXE(INDX)=INDX+1
      INDXW(INDX)=INDX-1
      Z=ZN(I)
      RHO=RNT(J)
      HN=0.
      HS=Z-ZN(I+1)
      HE=RNT(J+1)-RHO
      HW=RHO-RNT(J-1)
      ALPHA=ALPHAF(RHO,Z)
      CN=0.
      CS=0.5*(HE+HW)/HS*(RHO+(HE-HW)/4.)
      CE=0.5*HS/HE*(RHO+HE/2.)
      CW=0.5*HS/HW*(RHO-HW/2.)
      C=0.5*HS*(HE+HW)*(RHO/HE/HW+(1.-ALPHA*HS)/HS**2*(RHO+(HE-HW)/4.))
      V=0.25*HS*(HE+HW)*(RHO+(HE-HW)/4.)
    5 CALL PRINT
C
C      LAST POINT FIRST LINE
C
      J=JJN
      INDX=J
      INDXN(INDX)=0
      INDXS(INDX)=J+JJN
      INDXE(INDX)=0
      INDXW(INDX)=INDX-1
      INDXE(INDX)=0
      INDXW(INDX)=INDX-1
      INDXE(INDX)=0
      INDXW(INDX)=INDX-1
```



```
      Z=ZN(I)
      RHO=RNT(J)
      HN=0.
      HS=Z-ZN(I+1)
      HE=0.
      ALPHA=ALPHAF(RHO,Z)
      BETA=BETAF(RHO,Z)
      HW=RHO-RNT(J-1)
      CN=0.
      CS=0.5*HW/HS*(RHO-HW/4.)
      CE=0.
      CW=0.5*HS/HW*(RHO-HW/2.)
      C=0.5*((HW/HS+HS/HW-ALPHA*HW)*(RHO-HW/4.)-HS*(BETA*RHO+ 0.25))
      V=0.25*HS*HW*(RHO-HW/4.)
      CALL PRINT
C
C      MIDDLE LINES
C
      DO 10 I=2,NROWSN
      IF (IFIRST.EQ.0) WRITE (M,334) I
      DO 10 J=1,JJN
      INDX=J+(I-1)*JJN
      INDXN(INDX)=INDX-JJN
      INDXS(INDX)=INDX+JJN
      Z=ZNT(I)
      HN=ZNT(I-1)-Z
      HS=Z-ZNT(I+1)
      RHO=RNT(J)
      JGO=2
      IF(J.EQ.1) JGO=1
      IF(J.EQ.JJN) JGO=3
      GO TO (6,7,8),JGO
    6 INDXE(INDX)=INDX+1
      INDXW(INDX)=0.
      HE=RNT(J+1)-RHO
      HW=0.
      CALL LEFT(RHO,HN,HS,HE,HW,CN,CS,CE,CW,C,V)
      GO TO 9
    7 INDXE(INDX)=INDX+1
      INDXW(INDX)=INDX-1
      HE=RNT(J+1)-RHO
      HW=RHO-RNT(J-1)
      CALL MIDDLE(RHO,HN,HS,HE,HW,CN,CS,CE,CW,C,V)
      GO TO 9
    8 INDXE(INDX)=0
      INDXW(INDX)=INDX-1
      HE=0.
      HW=RHO-RNT(J-1)
      BETA=BETAF(RHO,Z)
      CALL RIGHT (RHO,HN,HS,HE,HW,BETA,CN,CS,CE,CW,C,V)
    9 CALL PRINT
```

```
   10 CONTINUE
C
C
C     Z = 0   PLANE
C
C
      IF(IFIRST.EQ.0) WRITE(M,444)
  444 FORMAT(/////1X, 9HZ=0 PLANE ///)
      I=1
      IF(IFIRST.EQ.0) WRITE(M,334) I
      DO 26 J=1,JJN
      INDX=J+NROWSN*JJN
      INDXN(INDX)=INDX-JJN
      INDXS(INDX)=INDX+JJS
      RHO=RNT(J)
      Z=0.
      JGO=2
      IF(J.EQ.1)JGO=1
      IF(J.EQ.JJN) JGO=3
      HN=ZN(NROWSN)
      HS=-ZS(1)
      GO TO (20,21,22),JGO
   20 INDXE(INDX)=INDX+1
      INDXW(INDX)=0
      JSAT=J
      HE=RNT(J+1) - RHO
      HW=0.
      CALL LEFT(RHO,HN,HS,HE,HW,CN,CS,CE,CW,C,V)
      CALL PRINT
      GO TO 26
   21 INDXE(INDX)=INDX+1
      INDXW(INDX)=INDX-1
      HE=RNT(J+1) - RHO
      HW=RHO - RNT(J-1)
      GO TO 23
   22 INDXE(INDX)=0
      INDXW(INDX)=INDX-1
      HE=0.
      HW=RHO - RNT(J-1)
      BETA=BETAF(RHO,Z)
      GO TO 24
   23 CALL MIDDLE(RHO,HN,HS,HE,HW,CN,CS,CE,CW,C,V)
      CALL PRINT
      GO TO 26
   24 CALL RIGHT(RHO,HN,HS,HE,HW,BETA,CN,CS,CE,CW,C,V)
      CALL PRINT
   26 CONTINUE
C
C
C     SOUTH + SOUTHEAST REGION
C
```



```
C
      IF(IFIRST.EQ.0) WRITE(M,555)
  555 FORMAT(/////25H SOUTH + SOUTHEAST REGION///)
      DO 41 I=1,NROWSS
      IF(IFIRST.EQ.0) WRITE(M,334) I
      DO 41 J=1,JJS
      INDX=J+(I-1)*JJS+IIN*JJN
      RHO=RST(J)
      Z=ZST(I+1)
      IGO=2
      JGO=2
      IF(I.EQ.1) IGO=1
      IF(I.EQ.NROWSS) IGO=3
      IF(J.EQ.1) JGO=1
      IF(J.EQ.JJS) JGO=3
      GO TO (28,28,30), IGO
C
C     FIRST AND MIDDLE LINES
C
   28 INDXN(INDX)=INDX-JJS
      INDXS(INDX)=INDX+JJS
      HN=ZST(I)-Z
      HS=Z-ZST(I+2)
      GO TO 31
C
C     LAST LINE
C
   30 INDXN(INDX)=INDX-JJS
      INDXS(INDX)=0
      HN=ZST(I) - Z
      HS=0.
      ALPHA=-ALPHAF(RHO,Z)
   31 GO TO (34,35,36), JGO
C
C     FIRST POINT
C
   34 INDXE(INDX)=INDX+1
      INDXW(INDX)=0
      HE=RST(J+1)-RHO
      HW=0.
      IF(IGO.EQ.3) GO TO 33
      CALL LEFT(RHO,HN,HS,HE,HW,CN,CS,CE,CW,C,V)
      GO TO 40
C
C     MIDDLE POINTS
C
   35 INDXE(INDX)=INDX+1
      INDXW(INDX)=INDX-1
      HE=RST(J+1)-RHO
      HW=RHO-RST(J-1)
      IF(IGO.EQ.3) GO TO 33
```

```
      CALL MIDDLE(RHO,HN,HS,HE,HW,CN,CS,CE,CW,C,V)
      GO TO 40
C
C     LAST POINT
C
   36 INDXE(INDX)=0
      INDXW(INDX)=INDX-1
      HE=0.
      HW=RHO-RST(J-1)
      BETA=BETAF(RHO,Z)
      IF (ISO.EQ.3) GO TO 33
      CALL RIGHT(RHO,HN,HS,HE,HW,BETA,CN,CS,CE,CW,C,V)
      GO TO 40
   33 GO TO (37,38,39),JSO
C
C     LAST LINE, FIRST POINT
C
   37 CN=0.125*HE**2/HN
      CS=0.
      CE=HN/4.
      CW=0.
      C=0.125*HE*(HE/HN + 2.*HN/HE - ALPHA*HE)
      V=HN*HE**2/16.
      GO TO 40
C
C     LAST LINE, MIDDLE POINTS
C
   38 CN=0.5*(HE+HW)/HN*(RHO+(HE-HW)/4.)
      CS=0.
      CE=0.5*HN/HE*(RHO+HE/2.)
      CW=0.5*HN/HW*(RHO-HW/2.)
      C=0.5*HN*(HE+HW)*(RHO/HE/HW +  (1.-ALPHA*HN)*(RHO+(HE-HW)/4.)/HN**
     12)
      V=0.250*HN*(HE+HW)*(RHO + (HE-HW)/4.)
      GO TO 40
C
C     LAST LINE, LAST POINT
C
   39 CN=0.5*HW/HN*(RHO-HW/4.)
      CS=0.
      CE=0.
      CW=0.5*HN/HW*(RHO-HW/2.)
      C=0.5*((HW/HN+HN/HW-ALPHA*HW)*(RHO-HW/4.)- HN*(BETA*RHO + 0.25))
      V=0.25*HN*HW*(RHO-HW/4.)
   40 CALL PRINT
   41 CONTINUE
      IPRINT=0
      IF (IPRINT.EQ.0) GO TO 43
      IF(IFIRST.GT.0.AND.IT.EQ.0.AND.NPHOTO.GT.0.AND.DERYE.EQ.0.)
     1 WRITE(4,777)(N,PHAND(N),N=1,NTOT)
  777 FORMAT(//1X,54HRHAND ARRAY FOR INVERSE CAPACITANCE MATRIX CALCULAT
```



```
     1ION /(8(I4,1PE12.4)))
   43 CONTINUE
      IF(IFIRST.GT.0) CALL SEIDEL
      RETURN
      END
```

```
      SUBROUTINE LEFT(RHO,HN,HS,HE,HW,CN,CS,CE,CW,C,V)
      CN=0.125*HE**2/HN
      CS=CN*HN/HS
      CE=0.25*(HN+HS)
      CW=0.
      C=0.25*(HN+HS)*(1.+0.5*HE**2/HN/HS)
      V=HE**2*(HN+HS)/16.
      RETURN
      END
```

19


```
      SUBROUTINE MIDDLE (RHO,HN,HS,HE,HW,CN,CS,CE,CW,C,V)
      CN=0.5*(HE+HW)/HN*(RHO+(HE-HW)/4.)
      CS=CN*HN/HS
      CE=0.5*(HN+HS)/HE*(RHO+HE/2.)
      CW=0.5*(HN+HS)/HW*(RHO-HW/2.)
      C=0.5*(HN+HS)*(HE+HW)*(RHO/HE/HW+(RHO+(HE-HW)/4.)/HN/HS)
      V=0.25*(HN+HS)*(HE+HW)*(RHO+(HE-HW) /4.)
      RETURN
      END
```

20

BEST AVAILABLE COPY

```
      SUBROUTINE RIGHT (RHO,HN,HS,HE,HW,BETA,CN,CS,CE,CW,C,V)
      CN=0.5*HW/HN*(RHO-HW/4.)
      CS=CN*HN/HS
      CE=0.
      CW=0.5*(HN+HS)/HW*(RHO-HW/2.)
      C=0.5*(HN+HS)*(HW/HN/HS*(RHO-HW/4.)+(RHO-HW/2.)/HW - BETA*RHO)
      V=0.25*(HN+HS)*HW*(RHO-HW/4.)
      RETURN
      END
```



```
      SUBROUTINE PRINT
      COMMON JJN,IIN,JJS,IIS,NTOT,RNT(50),ZNT(50),RST(50),ZST(50),
     1 RZ( 500,2),PHIN(20,20),PHIS(20,20),RHAND( 500),IFIRST,M
      COMMON/FLD/NCOLSN,NCOLSE,NCOLSS,NROWSN,X(500),NROWSS,DEBYE,DEBYE2,
     1 RHON1(50),RHOE(50),RHOS1(50),ZN(50),    ZS(50),PHI(50),NGAP,NDISK
      COMMON/A/CN,CS,CE,CW,C,V,  INDX,JSAT,RHO,Z,INDXN( 500),INDXS( 500)
     1,INDXE( 500),INDXW( 500),CONST( 500,6)
      CP=0.
      CONST(INDX,1)= CN
      IF((INDXN(INDX).EQ.0 ).AND.(CN.NE.0.)) CP=CN
      CONST(INDX,2)= CS
      IF((INDXS(INDX).EQ.0 ).AND.(CS.NE.0.)) CP=CS
      CONST(INDX,3)= CE
      CONST(INDX,4)= CW
      IF((INDXW(INDX).EQ.0 ).AND.(CW.NE.0.)) CP=CW
C *********** TEMPORARY --- HELMHOLTZ EQUATION
      IF(DEBYE2.GT.0.) C=C+ V/(DEBYE2**2)
C *********** TEMPORARY --- HELMHOLTZ EQUATION
      CONST(INDX,5)= C
      CONST(INDX,6)= V
      IF (CP.GT.0..AND.DEBYE.GT.0.) RHAND(INDX) = RHAND(INDX)*V/DEBYE**2
     1 + CP*PHI(JSAT)
      IF (CP.EQ.0..AND.DEBYE.GT.0.) RHAND(INDX) = RHAND(INDX)*V/DEBYE**2
      IF (IFIRST.GT.0) GO TO 3
      WRITE(M,1)   INDX, INDXN(INDX), CONST(INDX,1), INDXW(INDX), CONST(
     1INDX,4), INDX,CONST(INDX,5), INDXE(INDX), CONST(INDX,3), INDXS(IN
     2DX), CONST(INDX,2), CONST(INDX,6)
    1 FORMAT( /   6H POINT,I4,     3H/C(,I4,2H)=,1PE10.4,3H/C(,I4,2H)=,
     1E10.4,  3H/C(,I4,2H)=,E10.4,  3H/C(,I4,2H)=,E10.4,   3H/C(,I4,2H)=,
     2E10.4,  5H/VOL=,E10.4)
      IF  (CP.NE.0.) WRITE (M,2)  JSAT, CP
    2 FORMAT(31H COEFFICIENT OF POTENTIAL NO. (,I3, 4H) IS,F10.5)
    3 RZ(INDX,1)=RHO
      RZ(INDX,2)=Z
      RETURN
      END
```

```
      SUBROUTINE SEIDEL
      COMMON JJN,IIN,JJS,IIS,NTOT,RNI(50),ZNI(50),RSI(50),ZSI(50),
     1 RZ( 500,2),PHIN(20,20),PHIS(20,20),RHAND( 500),IFIRST,M
      COMMON/FLD/NCOLSN,NCOLSE,NCOLSS,NROWSN,X(500),NROWSS,DEBYE,DEBYE2,
     1 PHON1(50),RHOE(50),PHOS1(50),ZN(50),   ZS(50),PHI(50),NGAP,NDISK
      COMMON/A/CN,CS,CE,CW,C,V,  INDX,JSAT,RHO,Z,INDXN( 500),INDXS( 500)
     1,INDXE( 500),INDXW( 500),CONST( 500,6)
      COMMON/DEN1/ IT,NAME,NEWPHI,ISAVE,NGR,NGROUP( 500),DSAVE( 500)
     1 , ZFRONT,NSS,NROWS,NPHOTO,AREA(100),EFFLUX(100)
      N1=JJN*(IIN-1)
      NONCON=1
      RADIUS=RHON1(NCOLSN)
      OMEGA=1.3
      EPS= 0.00001
      ITMAX=2000
      ITCOUN =0
      IPPOLD=0
      IGO=1
      IF(NEWPHI.GT.0.AND.DEBYE.GT.0.) WRITE(M,100)
100   FORMAT(/1X,44HMODIFIED POISSON PROBLEM TO INCLUDE EXP(PHI))
      IF(IT.GT.0.AND.IFIRST.GT.0) GO TO 2
      DO 1 K =1,NTOT
    1 X(K)=0.
    2 ITCOUN =ITCOUN +1
      DELTAM=0.
      K1=0
      K2=0
      DO 3 K=1,NTOT
      X1=X(K)
      SN=CONST(K,1)/CONST(K,5)
      SS=CONST(K,2)/CONST(K,5)
      SE=CONST(K,3)/CONST(K,5)
      SW=CONST(K,4)/CONST(K,5)
      SR=RHAND(K)/CONST(K,5)
      INDXNK=INDXN(K)
      INDXSK=INDXS(K)
      INDXEK=INDXE(K)
      INDXWK=INDX4(K)
      AA=SR
      IF(INDXNK.GT.0) AA=AA+SN*X(INDXNK)
      IF(INDXSK.GT.0) AA=AA+SS*X(INDXSK)
      IF(INDXEK.GT.0) AA=AA+SE*X(INDXEK)
      IF(INDX4K.GT.0) AA=AA+SW*X(INDXWK)
      IF(NEWPHI.EQ.0.OR.DEBYE.EQ.0.) GO TO 30
C MODIFICATION TO INCLUDE EXP(PHI) IN POISSON PROBLEM
      BB=CONST(K,6)/CONST(K,5)/DEBYE**2
      CALL ROOT(AA,BB,XX)
      X(K)=XX
      GO TO 35
30    X(K)=AA
35    CONTINUE
```


BEST AVAILABLE COPY


```
C
C
********** MODIFICATION FOR FINITE DISK THICKNESS
C SET PHI TO BODY POTENTIAL AT ADDITIONAL BODY POINTS IF ANY.
      IF (RZ(K,1).GT.RADIUS.OR.RZ(K,2).LT.ZFRONT.OR.RZ(K,2).GT.0.)
     1 GO TO 20
      IF(IFIRST.GT.0.AND.IT.EQ.0.AND.NPHOTO.GT.0.AND.DEBYE.EQ.0.)
     1 GO TO 20
      IF(RZ(K,2).LT.0.) GO TO 6
      IF((K -N1).GE.1.AND.(K-N1).LE.NCOLSN) X(K)=PHI(K-N1)
       GO TO 20
6     CONTINUE
C     NO OVERRIDE IF NONCONDUCTING BODY
      IF (NONCON.GT.0) GO TO 20
      K1=K1+1
      NSS1K2=0
      INTARG=1+NCOLSN+K1/NCOLSS
      X(K)=0.
      IF (RZ(K,2).EQ.ZFRONT) GO TO 5
      MODFUN=MOD(K1,NCOLSS)
      IF (MODFUN.EQ.0) X(K)=PHI(INTARG)
      GO TO 20
5     K2=K2+1
      IF(K2.EQ.NCOLSS) X(K)=PHI(NSS+2)
      IF (K2.EQ.NCOLSS) GO TO 20
      NSS1K2=NSS+1-K2
      X(K)=PHI(NSS1K2)
C
20    CONTINUE
C     WRITE(M,1111) K1,K2,MODFUN,INTARG,NSS1K2
C     WRITE (M,1112) X(K)
1111  FORMAT (/1X33HK1, K2, MODFUN, INTARG, NSS1K2 = ,5I5)
1112  FORMAT (1X,5HX(K)=,1PE20.8  )
      X(K)=OMEGA*X(K)+(1.-OMEGA)*X1
      DELTA=ABS(X(K)-X1)
      IF(X1.NE.0.) DELTA=ABS((X(K)-X1)/X1)
      IF(DELTA.GT.DELTAM) DELTAM=DELTA
    3 CONTINUE
      IF(ITCOUN.GT.ITMAX) WRITE(M,11) ITMAX
      IF(ITCOUN.GT.ITMAX) GO TO 9
   11 FORMAT(////10H MORE THAN,I4,10HITERATIONS)
      IPP=ITCOUN/500
      IF(IPP.LE.IPPOLD) GO TO 15
      IPPOLD=IPP
      GO TO 10
   15 IF (DELTAM.GT.EPS) GO TO 2
C
C     ITERATION FINISHED
C
    9 IGO=2
   10 NEPP=(NTOT/300) + 1
```

```
      IF(IFIRST.GT.0.AND.IT.EQ.0.AND.NPHOTO.GT.0.AND.DERVE.EQ.0.)
     1 GO TO (15,4), IGO
      DO 51 IP=1,NEPP
      WRITE(M,22) ITCOUN,EPS,DELTAM,OMEGA
      DO 51 I=1,60
      K1=I + 300*(IP-1)
      K2=K1 + 60
      K3=K2 + 60
      K4=K3 + 60
      K5=K4 + 60
      IF(K5.LE.NTOT)WRITE(M,3333)K1,X(K1),K2,X(K2),K3,X(K3),K4,X(K4),
     1K5,X(K5)
      IF(K5.LE.NTOT) GO TO 51
      IF(K4.LE.NTOT)WRITE(M,3333)K1,X(K1),K2,X(K2),K3,X(K3),K4,X(K4)
      IF(K4.LE.NTOT) GO TO 51
      IF(K3.LE.NTOT)WRITE(M,3333)K1,X(K1),K2,X(K2),K3,X(K3)
      IF(K3.LE.NTOT) GO TO 51
      IF(K2.LE.NTOT)WRITE(M,3333)K1,X(K1),K2,X(K2)
       IF(K2.LE.NTOT) GO TO 51
      IF(K1.LE.NTOT)WRITE(M,3333)K1,X(K1)
   51 CONTINUE
 3333 FORMAT(5(I8,F16.8))
      GO TO (15,4),IGO
   22 FORMAT(15H1SOLUTION AFTER,I6,2X,25HITERATIONS WITH TOLERANCE,F12.8
     1,8X,18HMAXIMUM DIFFERENCE,F12.8,8X,6HOMEGA=,F8.5)
    4 RETURN
      END
```


BEST AVAILABLE COPY


```
      SUBROUTINE DENSTY
      COMMON JJN,IIN,JJS,IIS,ND,RN(50),ZN(50),RS(50),ZS(50),RSV( 500),
     1 ZSV( 500),PHIN(20,20),PHIS(20,20),CDSV( 500),IFIRST,M
      COMMON/DEN/NPRINT,MD,MC,MA,MB,ME,STEP,RSAVE,ZSAVE,ALPHA,BETA,FE,
     1 XMSAVE,RADIUS,FLUXI(3,20),FLUXE(3,20),FLUXEM(3,20),CDSAVE(100)
      COMMON/DEN1/IT,MAME,NEMPHT,ISAVE,NGR,NGROUP( 500),DSAVE( 500)
     1 , ZFRONT,NSS,NROWS,NPHOTO,AREA(100),EFFLUX(100)
      COMMON/PZINT,IGN,JGN,IGS,JGS
      COMMON/RZRZ/RA(50),RB(50),ZA(50),ZB(50),IIA,JJA,IIB,JJB
      COMMON/DRB/RDOT,ZDOT,AMS,NTIME,R1,R2,Z1,Z2,R,Z,PHI,RSAV,ZSAV
      COMMON/ZA/PI,XMACH,KEMAX,JEMAX,KBMAX,JBMAX,KAMAX,JAMAX,
     1 FEE(16,2),SNA(16,2),CSA(16,2),BTA(16,2),COFFA(16,2),COEFF(16,2)
      DIMENSION DNORTH(20,20),DSOUTH(20,20)
      DIMENSION PARTCL(2),PART1(2),PART2(2),FATE(2),END1(2),END2(2)
      DIMENSION FRAC(100),FLUX(100),ADFLUX(100),INDEX(30),KBHIT(30,50),
     1 JBHIT(30,50),KAHIT(30,50),JAHIT(30,50),NOSTEP(100)
      DATA PART1/6H ION   ,6H      /,PART2/6H ELECT,6HRON   /
      DATA END1/6H ABSOR,6HBED   /,END2/6H ESCAP,6HES    /
      PI = 3.1415926536
      INDXMX=50
      MSTEP=1000
      ROUND=1.E-12
      REPEAT=0.
      COMIN=0.
C     NORMALIZATION FACTOR FOR HYDROGEN ION FLUX
      FACTOR=1./SQRT(1836.)
      IPRINT=1
      IF(MC.GT.0.OR.NGR.GT.1     ) IPRINT=0
      NTOT=ND
      NPIS=ND
      NDP=ND+1
      IINM=IIN-1
      IISM=IIS-1
      N1=IINM*JJN
      N2=NTOT-IISM*JJS
      NCOLSN=(NSS-NROWS-1)/2
      IF (NROWS.EQ.0) NCOLSN=NSS
      NCOLSS=NCOLSN
C DO ONE CHARGE DENSITY OR CURRENT DENSITY, OR DO ALL
      IF(MD.EQ.0) NPIS=1
      IF(MD.EQ.0)  RSV(1)=RSAVE
      IF(MD.EQ.0)  ZSV(1)=ZSAVE
      IF(MC.EQ.0) WRITE(M,566) IT,NGR
      IF(MC.GT.0) WRITE(M,667) IT
      WRITE(M,664)NPRINT,MD,MC,MA,MB,ME,STEP,RSAVE,ZSAVE,
     1 ALPHA,BETA,FE,XMSAVE
  666 FORMAT(1H1/10H0DENSITIES,5X,4HIT =,I3,3X,5HNGR =,I3)
  667 FORMAT(1H1/9H0CURRENTS,5X,4HIT =,I3)
  664 FORMAT(1X,22HNPRINT,MD,MC,MA,MB,ME=,6I4/
     1 1X,37HSTEP,RSAVE,ZSAVE,ALPHA,BETA,EE,XMACH=,7F10.5/)
C
```

```
      IF(NPRINT.EQ.0) WRITE(M,660) NPRINT
      IF(NPRINT.EQ.1) WRITE(M,661) NPRINT
      IF(NPRINT.EQ.2) WRITE(M,662) NPRINT
      IF(NPRINT.EQ.3) WRITE(M,663) NPRINT
  660 FORMAT(1X,8HNPRINT =,I2,38H MEANS NO TRAJECTORY PRINTING          )
  661 FORMAT(1X,8HNPRINT =,I2,38H INDICES OF ESCAPING TRAJECTORIES ONLY)
  662 FORMAT(1X,8HNPRINT =,I2,38H=FIRST + LAST STEPS OF EACH TRAJECTORY)
  663 FORMAT(1X,8HNPRINT =,I2,38H EVERY STEP OF ALL TRAJECTORIES      )
333   FORMAT(/5H LINE,I4,5X,2HZ=,F8.4/(7F16.8))
3000  FORMAT(///,1X,24H POTENTIAL ARRAY - NORTH//1X,2HR=,16F8.4/
     1 (/3X,16F8.4))
4000  FORMAT(////24H POTENTIAL ARRAY - SOUTH//1X,2HR=,16F8.4/
     1 (/3X,16F8.4))
C
      WRITE (M,3000)(RN(J),J=1,JJN)
      DO 5 I=1,IIN
      WRITE(M,333) (I,ZN(I),(PHIN(I,J),J=1,JJN))
      DO 5 J=1,JJN
5     DNORTH(I,J)=0.
      WRITE(M,4000)(RS(J),J=1,JJS)
      DO 6 I=1,IIS
      WRITE(M,333) (I,ZS(I),(PHIS(I,J),J=1,JJS))
      DO 6 J=1,JJS
6     DSOUTH(I,J)=0.
      MOSTPS=0
C DO 95 LOOP ENDS AT END OF PROGRAM
      IF(MD.GT.0.AND.MC.EQ.0) NFTS=ND+1
      NSURF=0
      IF (MD.GT.0.AND.MC.GT.0) NPTS=ND+NSS
      IF (MD.GT.0.AND.MC.GT.0) NSURF=1
      NF1=NCOLSN
      NF2=NROWS+1
      NF3=NCOLSS
      NC1=ND+NCOLSN
      NC2=NC1+1
      NC3=NPTS-NCOLSS
      NC4=NC3+1
      IPMIN=1
      IPMAX=1
      IF (NPHOTO.EQ.1) IPMIN=2
      IF (NPHOTO.GT.0) IPMAX=2
      DO 9500 IP=IPMIN,IPMAX
      DO 95 N=1,NPTS
      IF (NSURF.GT.0) COSV(N)=0.
      IF (NSURF.GT.0.AND.N.LE.ND) GO TO 35
      IF (IP.EQ.2.AND.EFFLUX(N-ND).EQ.0.) GO TO 86
      IF (NPRINT.EQ.3)WRITE(M,3995)N
      RSAVE=RSV(N)
      ZSAVE=ZSV(N)
      NND=N-ND
      IF (NSURF.GT.0.AND.IFIRST.EQ.0) COSAVE(N-ND)=93.
```


BEST AVAILABLE COPY


```
      IN=0
      JN=0
      IS=0
      JS=0
      IF (NSURF.EQ.2.AND.NROWS.LE.0) NSURF=?
      IF (NSURF.GT.0) GO TO 19
      IF(N.LE.N1) GO TO 7
      IF(N.GT.N2) GO TO 8
7     DO 1 I=1,IIN
      IF(ZSAVE.EQ.ZN(I)) IN=I
1     CONTINUE
      DO 2 J=1,JJN
      IF(RSAVE.EQ.RN(J)) JN=J
2     CONTINUE
      IF(N.LE.N1) GO TO 9
8     DO 3 I=1,IIS
      IF(ZSAVE.EQ.ZS(I)) IS=I
3     CONTINUE
      DO 4 J=1,JJS
      IF(RSAVE.EQ.RS(J)) JS=J
4     CONTINUE
9     CONTINUE
      IF(N.LE.N2.AND.IN.GT.0.AND.JN.GT.0) DNORTH(IN,JN)=DSAVE(N)
      IF(N.GT.N1.AND.IS.GT.0.AND.JS.GT.0) DSOUTH(IS,JS)=DSAVE(N)
      IF(NGR.EQ.0) GO TO 15
      IF(IT.GT.0.AND.N.LT.NPTS.AND.NGROUP(N).NE.NGR) COSV(N)=DSAVE(N)
      IF(IT.GT.0.AND.N.LT.NPTS.AND.NGROUP(N).NE.NGR) GO TO 95
15    CONTINUE
********** MODIFICATION FOR FINITE DISK THICKNESS
      IF(RSAVE.LE.RADIUS.AND.ZSAVE.GE.ZFRONT.AND.ZSAVE.LT.0.) COSV(N)=0.
      IF(RSAVE.LE.RADIUS.AND.ZSAVE.GE.ZFRONT.AND.ZSAVE.LT.0.) GO TO 95
      IF(MC.EQ.0.AND.ISAVE.EQ.0) DSAVE(N)=0.
19    CONTINUE
      IF(NSURF.GT.0.AND.IFIRST.GT.0.AND.ABS(COSAVE(N-ND)).LT.COMIN)
     1 GO TO 96
      MASAVE=MA
      MBSAVE=MB
      MESAVE=ME
      STEPSV=STEP
      INCREA=0
      IF (IP.EQ.2) ME=0
      IF(MC.GT.0.OR.NAME.EQ.0) GO TO 20
***** INCREASE ACCURACY NEAR AXIS
      IF(RSAVE.LE.RN(2).AND.ZSAVE.GT.0.) MA=ME=16
      IF(RSAVE.LE.RN(2).AND.ZSAVE.GT.0.) STEP=.05
      IF(RSAVE.LE.RN(2).AND.ZSAVE.GT.0.) INCREA=1
20    CONTINUE
C FIRST WE DO THE IONS
      SCALE=1.
      PARTCL(1)=PART1(1)
      PARTCL(2)=PART1(2)
```

```
      IF (IP.EQ.2.AND.NPHOIO.EQ.1) SCALE=-1.
C RETURN FROM END OF MAIN FOR ELECTRONS
C
237   CONTINUE
      IF(SCALE.GT.0.) XMACH=XMSAVE
      IF(SCALE.LT.0.) XMACH=0.
      POWER=A3*S(XMACH)
      PHIMAX = 0.
      DO 11 I=1,IIN
      DO 10 J=1,JJN
   10 PHIN(I,J)=SCALE*PHIN(I,J)
      IF (PHIMAX.LT.PHIN(I,1)) PHIMAX = PHIN(I,1)
   11 CONTINUE
C
      DO 13 I=1,IIS
      DO 12 J=1,JJS
12    PHIS(I,J)=SCALE*PHIS(I,J)
   13 CONTINUE
C
C SET UP SUMS OVER TRAJECTORIES
C
      IF (MA.EQ.0) GO TO 32
      JAMAX=2
      JBMAX=2
      KAMAX=MA
      KBMAX=MB
      NUMBER=4A*MB*4
C DO ONLY ONE BETA ON AXIS (SYMMETRY)
      IF(RSAVE.EQ.0.) KBMAX=1
      IF(RSAVE.EQ.0.) NUMBER=MA*2
      IF(RSAVE.EQ.0.) JBMAX=1
      IF(SCALE.GT.0..AND.N.EQ.1) WRITE(M,668) MA,MB,NUMBER
  668 FORMAT(/1X,I6,16H ALPHA-INTERVALS,1X,I6, 15H BETA-INTERVALS,1X,I6,
     1 24H TRAJECTORIES PER ENERGY)
C
      IF(ME.EQ.0) GO TO 31
      JEMAX=2
      KEMAX=ME
      IF(SCALE.GT.0..AND.N.EQ.1) WRITE(M,670) ME
  670 FORMAT(1X,I6,47H ENERGY INTERVALS, WITH 2 ENERGIES PER INTERVAL//)
C
      GO TO 33
C
C SINGLE ENERGY
C
   31 JEMAX=1
      KEMAX=1
      IF(SCALE.GT.0..AND.N.EQ.1) WRITE(M,673) EE
  673 FORMAT(1X,31H MONOENERGETIC CASE WITH ENERGY,F10.5//)
C
      GO TO 33
```



```
C
C SINGLE TRAJECTORY ONLY
C
   32 JAMAX=1
      JBMAX=1
      JEMAX=1
      KAMAX=1
      KBMAX=1
      KEMAX=1
      NUMBER=1
      WRITE (M,669) ALPHA, BETA,EE
  669 FORMAT( 18H SINGLE TRAJECTORY/ 7H ALPHA=,F20.19, 8H DEGREES/
     1  6H BETA=,F20.18, 8H DEGREES/  8H ENERGY=,F10.6)
C
      ALPHA=ALPHA*PI/180.
      BETA=BETA*PI/180.
      WRITE( M,665) ALPHA,BETA
  665 FORMAT( 1X, 3HOR,/ 1X, 6HALPHA=,F11.9, 8H RADIANS/ 1X, 5HBETA=,F10
     1.8, 8H RADIANS)
C
      SINA=SIN(ALPHA)
      COSA=COS(ALPHA)
C
C SUM OVER ALPHA, BETA, AND ENERGY
C
C
33    CONTINUE
      CALL COABS
      DENST=0.
      NOSTEP(NVD)=0
      DO 339 NAR =1,NSS
      INDEX(NAR)=0
      FRAC(NAR)=0.
339   CONTINUE
      DO 1001 KE=1,KEMAX
      DO 1001 JE=1,JEMAX
      DENS=0.
      NOESC=0
      DO 1000 KB=1, KBMAX
      DO 1000 JB=1, JBMAX
      DO 1000 KA=1, KAMAX
      DO 1000 JA=1, JAMAX
C
C INITIAL POSITION
C
29    CONTINUE
      IF (IP.EQ.2.AND.IT.EQ.0) GO TO 1001
      R=RSAVE
      Z=ZSAVE
      X=R
      Y=0.
```

```
      INT=0
      CALL INTERP
      INT=1
      PHISAV=PHI
***** ASSUME BOLTZMANN FACTOR FOR DENSITY OF ELECTRONS (OVERRIDE)
      IF(ABS(PHI).GT.500.) GO TO 96
      IF(MC.EQ.0.AND.SCALE.LT.0.) DENSI=EXP(-PHI)
      IF(MC.EQ.0.AND.SCALE.LT.0.) GO TO 96
      IF(MC.EQ.0.AND.ISAVE.GT.0) DENSI=DSAVE(N)
      IF(MC.EQ.0.AND.ISAVE.GT.0) GO TO 96
C
      IF(STEP.LE.0.) WRITE(M,111)
111   FORMAT(/////1X,47HSTOP DUE TO STEP LE. ZERO ***** ***** *****)
      IF(STEP.LE.0.) STOP
C
C INITIAL VELOCITY
C
      SPEED=0.
      IF(MF.NE.0) GO TO 41
      E=EE
      IF (IP.EQ.2) E=1.+PHI
      IF(E.LT. PHI) WRITE(M,674) KE,JE,KB,JB,KA,JA,BETA1,ALPHA1,E,PHISAV
      IF(E.LT.PHI) GO TO 1001
      GO TO 40
41    CONTINUE
      E=EFE(KE,JE)+AMAX1(PHI,0.)
      IF(E.LT.0. ) WRITE(M,674) KE,JE,KB,JB,KA,JA,BETA1,ALPHA1,E,PHISAV
      IF(E.LT.0. ) STOP
C
   40 SPEED=SQRT(E - PHI)
      IF(MA.EQ.0) GO TO 39
      SINA=SNA(KA,JA)
      COSA=CSA(KA,JA)
      BETA=BTA(KB,JB)
      IF(R.EQ.0.) BETA=0.
C
   39 XDOT=SPEED*SINA*COS(BETA)
      YDOT=SPEED*SINA*SIN(BETA)
      ZDOT=SPEED*COSA
      IF (NSURF.EQ.3) ZDOT=-ZDOT
      IF (NSURF.EQ.2) ZDOT=-XDOT
      IF (NSURF.EQ.2) XDOT=SPEED*COSA
      RDOT=XDOT
      AMS=(R*YDOT)**2
C
      THETA=0.
      ATAN1=0.
      ATAN2=0.
      AMSRT=ABS(R*YDOT)
      IF(AMSRT.GT.0.) ATAN1=ATAN(R*RDOT/AMSRT)
C
```



```
      NTIME=-99
      NCHR=0
      NCHZ=0
      NCHMAX=5
      KSTEP=0
      ALPHA1 = ACOS(COSA)*180./PI
      BETA1 = BETA*180./PI
      ALPHA =ALPHA1
      BETA=BETA1
      ZOLD=Z
      IF(NPRINT.NE.2.AND.NPRINT.NE.3) GO TO 42
C PRINT INITIAL CONDITIONS OF TRAJECTORY
      WRITE(M,674) KE,JE,KB,JB,KA,JA,BETA1,ALPHA1,E,PHISAV
  674 FORMAT(1X,3(I3,I2),F17.8,F14.8,2X,1P2F11.3,2X,46H =KE,JE,  KB,JB,
     1 KA,JA,  BETA1,ALPHA1,  E,PHI)
C
      WRITE(M,559)
  559 FORMAT(12X,120HSTEPS    X             Y             Z           XDOT
     1 YDOT         ZDOT          RDOT          R          IGN  JGN  IGS  JGS NTIM
     2E  )
C
      WRITE(M,888) KSTEP,X,Y,Z,XDOT,YDOT,ZDOT,RDOT,R,IGN,JGN,IGS,JGS
     1,NTIME
888   FORMAT(13X,I4,1P8F11.3,I3,4I5)
   42 IF(KSTEP.EQ.0) GO TO 35
C
C TAKE A STEP
C
C
35    CALL ORBIT
      IF(NTIME.EQ.-1) WRITE(M,9999)
      IF(NTIME.EQ.-2) WRITE(M,9998)
      IF(NTIME.EQ.-3) WRITE(M,9997)
      IF(NTIME.LT.0) WRITE(M,9995)N
      IF(NTIME.LT.0) WRITE(M,674)KE,JE,KB,JB,KA,JA,BETA1,ALPHA1,E,PHISAV
      IF(NTIME.LT.0) WRITE(M,9996)R,RSAV,Z,ZSAV
      IF(NTIME.LT.0) WRITE(M,888)KSTEP,X,Y,Z,XDOT,YDOT,ZDOT,RDOT,R,IGN,JG
     1N,IGS,JGS,NTIME
      IF(NTIME.LT.0) STOP
9999  FORMAT(/////1X,16HCANNOT FIND TIME)
9998  FORMAT(/////1X,14HDR.GT.10*ROUND)
9997  FORMAT(/////1X,14HDZ.GT.10*ROUND)
9996  FORMAT(/////1X,14HR,RSAV,Z,ZSAV=,1P4E25.15)
9995  FORMAT(/1X,3HPOINT NO.,I5//)
      KSTEP=KSTEP+1
C
      IF(NPRINT.LT.2) GO TO 38
      IF(AMSRT.GT.0.) ATAN2=ATAN(R*RDOT/AMSRT)
      THETA=THETA + ATAN2 - ATAN1
      ATAN1=ATAN2
      COSTH=COS(THETA)
```

```
      SINTH=SIN(THETA)
      XDOT=RDOT*COSTH
      YDOT=RDOT*SINTH
      IF(R.GT.0.) XDOT=XDOT - (AMSRT/R)*SINTH
      IF(R.GT.0.) YDOT=YDOT + (AMSRT/R)*COSTH
      X=R*COSTH
      Y=R*SINTH
 39   CONTINUE
C
      IF(NPRINT.EQ.3)
     1WRITE(M,888) KSTEP,X,Y,Z,XDOT,YDOT,ZDOT,RDOT,R,IGN,JGN,IGS,JGS
     2,NTIME
      IF(KSTEP.LT.MSTEP) GO TO 35
      WRITE (M,999) MSTEP
      WRITE(M, 97) KSTEP,N,KE,JE,KB,JB,KA,JA
  999 FORMAT(10H MORE THAN,I10,6HSTEPS)
      IF (NPRINT.EQ.3.OR.REPEAT.GT.0.) STOP
      REPEAT=1.
      NPRINT=3
      GO TO 29
C
 35   PHIOLD=PHI
      RR=R
      ZZ=Z
      IF(RDOT.NE.0.) R=R + ROUND*SIGN(1.,RDOT)
      IF(ZDOT.NE.0.) Z=Z + ROUND*SIGN(1.,ZDOT)
      IF (R.LT.RADIUS.AND.SIGN(1.,Z).NE.SIGN(1.,ZOLD)) GO TO 36
********** MODIFICATION FOR FINITE DISK THICKNESS
      IF(R.LT.RADIUS.AND.Z.GE.ZFRONT.AND.Z.LT.0.) GO TO 36
      IF(R.GT.RN(JJN).OR.Z.GT.ZN(1).OR.Z.LT.ZS(IIS)) GO TO 37
      ZOLD=Z
      IF(R.LT.0..AND.RR.EQ.0.) RDOT=-RDOT
      IF(R.LT.0..AND.RR.EQ.0.) R=0.
      CALL INTERP
C
      R=RR
      Z=ZZ
      IF (KSTEP.EQ.0) GO TO 34
      VELSQ=PHIOLD-PHI
      IF(NTIME.LE.4) VELSQ=VELSQ + RDOT**2
      IF(NTIME.LE.4.AND.VELSQ.GE.0.) RDOT=SQRT(VELSQ)*SIGN(1.,RDOT)
      IF(NTIME.GT.4) VELSQ=VELSQ + ZDOT**2
      IF(NTIME.GT.4.AND.VELSQ.GE.0.) ZDOT=SQRT(VELSQ)*SIGN(1.,ZDOT)
C
      IF(VELSQ.GE.0.) GO TO 30
      IF(NTIME.LE.4) RDOT=-RDOT
      IF(NTIME.LE.4) NCHR=NCHR+1
      IF(NTIME.LE.4.AND.RDOT.NE.0.)R=R+ROUND*SIGN(1.,RDOT)
      IF(NTIME.GT.4) ZDOT=-ZDOT
      IF(NTIME.GT.4) NCHZ=NCHZ+1
      IF(NTIME.GT.4.AND.ZDOT.NE.0.)Z=Z+ROUND*SIGN(1.,ZDOT)
```



```
C
      IF(NCHR.LT.NCHMAX.OR.NCHZ.LT.NCHMAX) GO TO 25
      WRITE (4,777) NCHR,NCHZ,NCHMAX
  777 FORMAT(/1X,5HNCHR=,I3,10H AND NCHZ=,I3, 42H HAVE ATTAINED THE MAXI
     1MUM ALLOWED NUMBER=,I3, 30H AND THE TRAJECTORY IS ABORTED)
      WRITE(M,97) KSTEP,N,KE,JE,KB,JB,KA,JA
      WRITE(M,888) KSTEP,X,Y,Z,XDOT,YDOT,ZDOT,RDOT,R,IGN,JGN,IGS,JGS,
     1 NTIME
      GO TO 1000
 25   CONTINUE
C
      CALL INTERP
      R=RR
      Z=ZZ
 30   CONTINUE
C
      IF(NPRINT.LT.2) GO TO 34
      IF(AMSRT.GT.0.) ATAN2=ATAN(R*RDOT/AMSRT)
      ATAN1=ATAN2
      COSTH=COS(THETA)
      SINTH=SIN(THETA)
      XDOT=RDOT*COSTH
      YDOT=RDOT*SINTH
      IF(R.GT.0.) XDOT=XDOT - (AMSRT/R)*SINTH
      IF(R.GT.0.) YDOT=YDOT + (AMSRT/R)*COSTH
      X=R*COSTH
      Y=R*SINTH
      IF(NPRINT.EQ.3)
     1WRITE(M,888) KSTEP,X,Y,Z,XDOT,YDOT,ZDOT,RDOT,R,IGN,JGN,IGS,JGS
     2,NTIME
C
      GO TO 34
C
C PARTICLE IS ABSORBED
C
   36 CONTINUE
      R=RR
      Z=ZZ
      IF (NSURF.EQ.0.OR.IP.LT.2) GO TO 360
      CALL INTERP
      NAR=0
      IF (Z.EQ.0.) NAR =JGN
      IF (R.EQ.RADIUS) NAR =NCOLGN+IGS
      IF (Z.EQ.ZFRONT) NAR =NSS+1-JGS
      INDEX(NAR)=INDEX(NAR)+1
      INDEXX=INDEX(NAR)
      IF (INDEXX.GT.INDXMX) WRITE(M,355) NAR,NND,INDEXX
      IF(INDEXX.GT.INDXMX)WRITE(M,7712)NND,INDEXX,NAR,(KORB,KBHIT(NAR,KO
     1RB),JBHIT(NAR,KORB),KAHIT(NAR,KORB),JAHIT(NAR,KORB),KORB=1,INDEXX)
      IF (INDEXX.GT.INDXMX) STOP
  355 FORMAT(/////1X,*INDEX OF ORBITS HITTING AREA*,I3,* FROM AREA*,I4,
```


BEST AVAILABLE COPY

```
     1" TS",I3," WHICH EXCEEDS ALLOWED DIMENSIONS")
      KAHIT(NAR,INDEXX)=KD
      JDHIT(NAR,INDEXX)=JD
      KAHIT(NAR,INDEXX)=KA
      JAHIT(NAR,INDEXX)=JA
      FRAC(NAR)=FRAC(NAR)+1./FLOAT(NUMBER)
  360 CONTINUE
      IF(NPRINT.NE.2.AND.NPRINT.NE.3) GO TO 1002
      FATE(1)=END1(1)
      FATE(2)=END1(2)
      GO TO 374
C
C PARTICLE ESCAPES
C
 37   IF(NPRINT.EQ.1) GO TO 372
      IF(NPRINT.NE.2.AND.NPRINT.NE.3) GO TO 373
      FATE(1)=END2(1)
      FATE(2)=END2(2)
      GO TO 373
372   WRITE(M,674) KE,JE,KB,JB,KA,JA,BETA1,ALPHA1,E,PHISAV
373   NOESC=NOESC+1
      IF(ME.EQ.0) GO TO 374
C
      CSANGL=ZDOT/SQRT(E-PHI)
      XPON=-2.*XMACH*SQRT(E)*CSANGL - E - XMACH**2
      IF(MC.EQ.0) COEFF1=COEFA(KA,JA)*SPEED/FLOAT(NUMBER)
      IF(MC.GT.0) COEFF1=SPEED**2/FLOAT(NUMBER)
      IF(ABS(XPON).GT.500.) GO TO 374
      DADD=COEFF1*EXP(XPON)
      DENS=DENS + DADD
C
  374 IF(NPRINT.NE.2.AND.NPRINT.NE.3) GO TO 1002
      WRITE(M,899) FATE,KSTEP,X,Y,Z,XDOT,YDOT,ZDOT,RDOT,R,IGN,JGN,IGS,JGS
     1,NTIME
899   FORMAT(1X,2A6,I4,10E11.3,I3,4I5)
C
 1002 CONTINUE
      IF(NOSTEP(NND).LT.KSTEP) NOSTEP(NND)=KSTEP
      IF(NOSTPS.GE.KSTEP) GO TO 1000
      KES=KE
      JES=JE
      KBS=KB
      JBS=JB
      KAS=KA
      JAS=JA
      NSAVE=N
      NOSTPS=KSTEP
 1000 CONTINUE
C
C END OF ANGLE SUM
C
```



```
      FRACT=FLOAT(NOESC)/FLOAT(NUMBER)
C
      IF(NPRINT.GT.0.OR.(MD.EQ.0.AND.MC.GT.0))
     1                          WRITE(M,671)NOESC,NUMBER,FRACT,E,DENS
  671 FORMAT(1 "HG RATIO ESCAPING =, I5, 8H OUT OF ,I5,
     1  15H OR A FRACTION ,F13.8,     13H AT ENERGY E=,F13.8,4X,
     2  6HIDENS=,1PE13.4,1H))
C
      IF(NPRINT.EQ.0) GO TO 5555
      IF(ME.NE.0.AND.MC.EQ.0) WRITE(M,675)
      IF(ME.NE.0.AND.MC.GT.0) WRITE(M,676)
675   FORMAT(1X, 66HDENS IS THE SUM OF DADD=SPEED*EXP(XPON)/NUMBER OVER
     1ALL DIRECTIONS//)
676   FORMAT(1X, 67HDENS IS THE SUM OF DADD=SPEED**2*EXP(XPON)/NUMBER OV
     1ER A HEMISPHERE//)
C
 5555 IF(ME.EQ.0) GO TO 1001
      COEFF2=COEFE(KE,JE)
      DENST=DENST + COEFF2*DENS
C
 1001 CONTINUE
C
      IF(NSURF.EQ.0.OR.IPRINT.LT.2) GO TO 83
      NOHITS=0
      DO 87 NAR=1,NSS
      ADFLUX(NAR)=FRAC(NAR)*AREA(N-ND)*EFFLUX(N-ND)/AREA(NAR)
      FLUX(NAR)=FLUX(NAR) + ADFLUX(NAR)
      IF(INDEX(NAR).EQ.0) GO TO 87
      NOHITS=1
      INDEXX=INDEX(NAR)
      WRITE(M,7712)NND,INDEXX,NAR,(KORB,KBHIT(NAR,KORB),JBHIT(NAR,KORB),
     1 KAHIT(NAR,KORB),JAHIT(NAR,KORB), KORB=1,INDEXX)
   87 CONTINUE
      IF (NOHITS.EQ.0) WRITE(M,7713)
      DENST=0.
C
      WRITE(M,7710) NND,AREA(NND),EFFLUX(NND)
      WRITE(M,7711) (NAR,FRAC(NAR),ADFLUX(NAR),FLUX(NAR), NAR=1,NSS)
7710  FORMAT(//1X,17HFROM AREA NUMBER ,I3,6X,20HAREA(N), EFFLUX(N) =,
     1 1P2E20.8//5X,32HN, FRAC(N), ADFLUX(N),FLUX(N) = )
7711  FORMAT( 7(1X,I5,1P3E12.4))
7712  FORMAT(//1X,13HFROM AREA NO.,I4,1H,3X,I3,18H  HITS ON AREA NO.,I4,
     1 32H ARE MADE BY ORBITS WITH INDICES /
     2(20X,19HK, KB, JB, KA, JA =, I3,5X,2I3,4X,2I3))
7713  FORMAT(////1X,21H-----  NO HITS  -----      )
C
      GO TO 90
   83 CONTINUE
      IF(ME.EQ.0.AND.MC.EQ.0) DENST=SPEED*FRACT
      IF(ME.EQ.0.AND.MC.GT.0) DENST=SPEED**2*FRACT
35    IF(MC.GT.0) WRITE(M,677) PSAVE,ZSAVE,PHISAV,PARTCL,DENST
```


BEST AVAILABLE COPY

```
  677 FORMAT(26H AT R=,F13.8, 7H AND Z=,F13.8, 19H, THE POTENTIAL IS=,
     1E13.3/1X,20H AND THE NORMALIZED ,2A6,20H CURRENT DENSITY IS=,1PE13
     2.4//)
C
      IF(NSURF.EQ.0) GO TO 89
      IF (NSURF.EQ.1) NFSF=N-ND
      IF (NSURF.EQ.2) NFSF=N-ND-NF1
      IF (NSURF.EQ.3) NFSF=NPTS-N+1
      IF(SCALE.GT.0.)FLUXI(NSURF,NFSF)=DENST*FACTOR
      IF(SCALE.LT.0.)FLUXE(NSURF,NFSF)=DENST
   89 CONTINUE
      IF(SCALE.LT.1..AND.MC.EQ.0) GO TO 91
      IF (SCALE.LT.0..AND.NSURF.GT.0) GO TO 91
      IF(SCALE.LT.1..AND.MC.GT.0) GO TO 90
      SCALE=-1.
      PARTCL(1)=PART2(1)
      PARTCL(2)=PART2(2)
      DENSA=DENST
      IF(NSURF.GT.0) GO TO 237
      IF(MC.EQ.0.AND.ISAVE.EQ.0) DSAVE(N)=DENSA
      IF(N.LE.N2.AND.IN.GT.0.AND.JN.GT.0) DNORTH(IN,JN)=DENSA
      IF(N.GT.N1.AND.IS.GT.0.AND.JS.GT.0) DSOUTH(IS,JS)=DENSA
      IF(MD.GT.0.AND.MC.EQ.0.AND.N.EQ.NPTS) DNORTH(IIN,1)=DENSA
      GO TO 237
C RETURN TO BEGINNING OF TRAJECTORIES FOR ELECTRONS
C
C CONTINUE IF IONS AND ELECTRONS COMPLETED
   91 CD=DENSA-DENST
      IF(NSURF.GT.0) CD=FACTOR*DENSA-DENST
      IF (NSURF.GT.0) CDSAVE(N-ND)=CD
      CDSV(N)=CD
      IF(MC.EQ.0.AND.NEWPHI.EQ.0) DSAVE(N)=CD
C SAVE ION DENSITY ONLY IF EXP(PHI) IS TO BE INCLUDED IN POISSON SOLUTIO
      IF(MC.EQ.0.AND.NEWPHI.GT.0) CDSV(N)=DENSA
      PHISAV=-PHISAV
      IF(MC.EQ.0) WRITE(M,672) N,RSAVE,ZSAVE,PHISAV,DENSA,DENST,CD
      IF (NSURF.GT.0)WRITE(M,672) N,RSAVE,ZSAVE,PHISAV,DENSA,DENST,CD
  672 FORMAT(1X, 5HAT N=,I4,9H R,Z,PHI=,1P2E10.2,E12.4,
     1 40H, THE ION/ELECTRON/CHARGE DENSITIES ARE=, 3E13.4)
      IF(INCREA.GT.0) WRITE(M,6721) N,MA,MB,MF,STEP
 6721 FORMAT(1X,12HFOR POINT N=, I3,4X,
     1 33HMA,MB,ME, AND STEP ARE CHANGED TO,3I5,4H AND,F10.5,
     2 23H FOR INCREASED ACCURACY/)
C
   90 DO 92 I=1,IIN
      DO 92 J=1,JJN
   92 PHIN(I,J)=SCALE*PHIN(I,J)
      DO 93 I=1,IIS
      DO 93 J=1,JJS
   93 PHIS(I,J)=SCALE*(PHIS(I,J))
***** RESTORE MA,MB,ME,STEP
```



```
      MA=MASAV
      MB=MBSAVE
      MF=MFSAVE
      STEP=STEPSV
   86 CONTINUE
      IF(NSURF.EQ.1.AND.RSAVE.EQ.RADIUS.AND.ZSAVE.EQ.0.) NSURF=2
      IF(NSURF.EQ.2.AND.RSAVE.EQ.RADIUS.AND.ZSAVE.EQ.ZFRONT) NSURF=3
   95 CONTINUE
      IF (IP.LT.2) GO TO 9500
      DO 940 N=1,NPTS
      IF (N.LE.ND) GO TO 940
      NND=N-ND
      IF(NSURF.GT.0.AND.IFIRST.GT.0.AND.ABS(CDSAVE(N-ND)).LT.CDMIN)
     1 GO TO 940
      CDSV(N)=CDSV(N)+FEFLUX(N-ND)-FLUX(N-ND)
      IF (NSURF.GT.0) CDSAVE(N-ND)=CDSV(N)
      IF(N.LE.(ND+NF1)) GO TO 941
      IF (N.GT.(ND+NF1+NF2)) GO TO 943
      GO TO 942
  941 NSURF=1
      NFSF=N-ND
      GO TO 944
  942 NSURF=2
      NFSF=N-ND-NF1
      GO TO 944
  943 NSURF=3
      NFSF=NPTS+1-N
  944 FLUXEN(NSURF,NFSF)=FLUX(N-ND)
  940 CONTINUE
 9500 CONTINUE
C
C     END OF N-LOOP
C
      IF(MD.GT.0) WRITE(M,666) IT,NGR
      IF(NSURF.EQ.0)WRITE(M,94)(N,RSV(N),ZSV(N),CDSV(N),N=1,NPTS)
      IF(NSURF.EQ.0) GO TO 85
      CORN1=.5*(CDSV(NC1)+CDSV(NC2))
      CORN2=.5*(CDSV(NC3)+CDSV(NC4))
      CDSV(NC1)=CORN1
      CDSV(NC2)=CORN1
      CDSV(NC3)=CORN2
      CDSV(NC4)=CORN2
      CDSAVE(NC1-ND)=CORN1
      CDSAVE(NC2-ND)=CORN1
      CDSAVE(NC3-ND)=CORN2
      CDSAVE(NC4-ND)=CORN2
   85 CONTINUE
      IF(NSURF.GT.0)WRITE(M,9401)(N,RSV(N),ZSV(N),CDSV(N),CDSAVE(N-ND),
     1 NOSTEP(N-ND),N=NDF,NPTS)
 9401 FORMAT(/4X,1HN,6X,1HR,12X,1HZ,12X,18HNET CURRENT DENSITY,5X,10HMOST
     1 STEPS //(1X,I4,0P2F13.5,1P2E13.4,5X,I5 ))
```

```
96    FORMAT(/5X,1HN,8X,1HR,12X,1HZ,12X,4HD/CO/(1X,I4,3F13.5))
C TRAJECTORY WITH MOST STEPS. PRINT INDICES(N,AND K AND J INDICES)
                        WRITE(M,97) MOSTPS,NSAVE,KFS,JFS,KBS,JBS,KAS,JAS
   97 FORMAT(1X,I5,I4, 3(I3,I2),34H =MOSTSTEPS,N, KF,JF, KB,JB, KA,JA )
      IF(NSURF.GT.0) IPRINT=1
      IF(NSURF.GT.0) GO TO 98
      IF(IPRINT.EQ.0) GO TO 99
      WRITE(M,2001)
      IF(ISAVE.GT.0) WRITE(M,2003)
2001  FORMAT(1H1/1X,21HDENSITY ARRAY - NORTH  )
      WRITE(M,2004) (RN(J),J=1,JJN)
2004  FORMAT(/1X,2HR=,16F8.4/(/3X,16F8.4))
2003  FORMAT(1X,40HDENSITIES READ IN RATHER THAN CALCULATED //)
      DO 100 I=1,IIN
      WRITE(M,333) I,ZN(I),(DNORTH(I,J),J=1,JJN)
100   CONTINUE
      WRITE(M,2002)
2002  FORMAT(////1X,21HDENSITY ARRAY - SOUTH//)
      WRITE(M,2004) (RS(J),J=1,JJS)
      DO 101 I=1,IIS
      WRITE(M,333) I,ZS(I),(DSOUTH(I,J),J=1,JJS)
101   CONTINUE
      GO TO 99
98    CONTINUE
      WRITE(M,2005)
2005  FORMAT(1H1/1X,21HCURRENT DENSITY ARRAY)
      WRITE(M,2004) (RN(J),J=1,NF1)
      ZOUT=0.
      WRITE(M,334) ZOUT,(FLUXI(1,J),J=1,NF1)
      WRITE(M,335)      (FLUXE(1,J),J=1,NF1)
      WRITE(M,339) (FLUXEM(1,J),J=1,NF1)
334   FORMAT(//1X,2HZ=,F8.4,7H (IONS),14F8.4/(18X,14F8.4))
335   FORMAT(6X,12H (ELECTRONS),14F8.4/(18X,14F8.4))
  338 FORMAT(6X,12H  (EMISSION),14F8.4/(18X,14F8.4))
      DO 102 I=1,NF2
      IF (NF2.EQ.0) GO TO 102
      ZOUT=ZS(I)
      WRITE(M,336) ZOUT,FLUXI(2,I)
      WRITE(M,337)      FLUXE(2,I)
      WRITE(M,339)FLUXEM(2,I)
102   CONTINUE
336   FORMAT(//1X,12HR=RADIUS, Z=,F8.4,84X,7H (IONS),F8.4)
337   FORMAT(100X,12H (ELECTRONS),F8.4)
  339 FORMAT(100X,12H  (EMISSION),F8.4)
      WRITE(M,2004) (RS(J),J=1,NF3)
      ZOUT=ZFRONT
      WRITE(M,334) ZOUT, (FLUXI(3,J), J=1,NF3)
      WRITE(M,335)       (FLUXE(3,J), J=1,NF3)
      WRITE(M,338) (FLUXEM(3,J),J=1,NF3)
99    CONTINUE
      RETURN
```



```
      END
```

40

```
      SUBROUTINE COAPS
C
C     PRECALCULATION OF COEFFICIENTS AND ABSCISSAS FOR ENERGY AND ANGLE
C     QUADRATURES
C
      COMMON JIN,IIN,JJS,IIS,ND,RN(50),ZN(50),RS(50),ZS(50),PSV(500),
     1 ZSV( 500),PHIM(20,20),PHIS(20,20),COSV( 500),IFIRST,M
      COMMON/DEN/NPRINT,MD,MC,MA,MU,ME,STEP,RSAVE,ZSAVE,ALPHA,BETA,EE,
     1 XMSAVE,RADIUS,FLUXI(3,20),FLUXE(3,20),FLUXEM(3,20),CPSAVE(100)
      COMMON/CA/PI,XMACH,KEMAX,JEMAX,KBMAX,JBMAX,KAMAX,JAMAX,
     1 EEE(16,2),SNA(16,2),CSA(16,2),BTA(16,2),COEFA(16,2),COEFF(16,2)
      DIMENSION A(2)
      A(1)=-1./SORT(3.)
      A(2)=-A(1)
      POWER=ABS(XMACH)
      XM=1.
      DO 100 KE=1,KEMAX
      DO 100 JE=1,JEMAX
      CE=(A(JE) + FLOAT(2*KE-1-KEMAX))/FLOAT(KEMAX)
      E=(1.+CE)/(1.-CE)
      IF(POWER.GT.XM) E=POWER**2*E
      EEE(KE,JE)=E
C
      IF(MC.GT.0) CCE=2./(1.-CE)**2/FLOAT(KEMAX)
      IF(MC.EQ.0) CCE=CCE*2./SORT(PI)
      IF(POWER.GT.XM) CCE=CCE*XMACH**2
C
      COEFE(KE,JE)=CCE
C
100   CONTINUE
C
      DO 300 KB=1,KBMAX
      DO 300 JB=1,JBMAX
      CBETA=(A(JB) + FLOAT(2*KB-1-KBMAX))/FLOAT(KBMAX)
      BETA=PI*(1.+CBETA)/2.
      BTA(KB,JB)=BETA
300   CONTINUE
C
      DO 200 KA=1,KAMAX
      DO 200 JA=1,JAMAX
      CA=(A(JA) + FLOAT(2*KA-1-KAMAX))/FLOAT(KAMAX)
      IF(MC.EQ.0) COSA=CA
C
C     FOR LARGE MACH NO. EMPHASIZE VELOCITY DIRECTIONS TOWARD UPWIND
C
      IF(MC.EQ.0.AND.XMACH.GT. XM) COSA=-1. + 2.*((1.+CA)/2.)**POWER
      IF(MC.EQ.0.AND.XMACH.LT.-XM) COSA= 1. - 2.*((1.-CA)/2.)**POWER
C
      IF(MC.EQ.0) SINA=SORT(1.-COSA**2)
C
      IF(MC.GT.0) SINA=SORT(.5*(1.+CA))
```



```
      IF(MC.GT.0) COSA=SORT(1.-SINA**2)
C
      SNA(KA,JA)=SINA
      CSA(KA,JA)=COSA
C
      CCA=1.
      IF(MC.EQ.0.AND.XMACH.GT. XM)CCA=CCA*POWER*((1.+CA)/2.)**(POWER-1.)
      IF(MC.EQ.0.AND.XMACH.LT.-XM)CCA=CCA*POWER*((1.-CA)/2.)**(POWER-1.)
      COEFA(KA,JA)=CCA
200   CONTINUE
      WRITE(M,700) ((I,J,EEE(I,J),SNA(I,J),CSA(I,J),BTA(I,J),COEFA(I,J),
     1COEFE(I,J),J=1,JEMAX),I=1,KEMAX)
700   FORMAT(/20X,3HEEE,11X,3HSNA,11X,3HCSA,11X,3HBTA,10X,5HCOEFA,10X,
     15HCOEFE/(1X,I4,I6,3X,1P6E14.5/ 1X,I4,I6,3X,1P6E14.5//))
      RETURN
      END
```

```
      SUBROUTINE INTERP
      COMMON JJA,ITA,JJB,IIB,ND,RA(50),ZA(50),RB(50),ZB(50),RSV( 500),
     1 ZSV( 500),PHIN(20,20),PHIS(20,20),CISV( 500),IFIRST,H
      COMMON/P/INT,IGN,JGN,IGS,JGS
      COMMON/R7Z/RN(50),RS(50),ZN(50),ZS(50),IIN,JJN,IIS,JJS
      COMMON/D2B/RDOT,ZDOT,ANS,NTIME,R1,R2,Z1,Z2,R,Z,PHI,RSAV,ZSAV
      IGN=JGN=IGS=JGS=0
      NCH=0
      IGO2 = 0
      IF(INT.NE.0) IGO2 = IGO
      IGO=2
      IF (Z.GE.0.) IGO=1
      IF(IGO.NE.IGO2) INT = 0
      GO TO (1,2),IGO
C
C     NORTH Z
C
C     ASSUMING ZERO LESS THAN OR EQUAL TO Z LESS THAN OR EQUAL TO ZN(IIN
C
    1 IF(Z.EQ.ZN(1)) IG=2
      IF(Z.EQ.ZN(1)) GO TO 103
      IF(INT.NE.0) GO TO 100
      DO 10 I=2,IIN
      IG=IIN-I+2
      IF(Z.LT.ZN(IG-1)) GO TO 103
   10 CONTINUE
C
C     ACCEPT IF ZN(IG) LESS THAN OR EQUAL TO Z LESS THAN ZN(IG-1).
C
  100 IF(Z.GE.ZN(IG-1)) GO TO 102
      IF(Z.GE.ZN(IG)) GO TO 104
  101 IG=IG+1
      IF(Z.LT.ZN(IG)) GO TO 101
      GO TO 103
  102 IG=IG-1
      IF(Z.GE.ZN(IG-1)) GO TO 102
  103 NCH=1
  104 CONTINUE
C
C     NORTH R
C
C     ASSUMING RN(1) LESS THAN OR EQUAL TO R LESS THAN OR EQUAL TO RN(JJ
C
      IF(R.EQ.RN(JJN)) JG=JJN-1
      IF(R.EQ.RN(JJN))GO TO 153
      IF(INT.NE.0) GO TO 150
      DO 15 J=2,JJN
      JG=J-1
      IF(R.LT.RN(J)) GO TO 153
   15 CONTINUE
C
```



```
C     ACCEPT IF RN(JG) LESS THAN OR EQUAL TO R LESS THAN RN(JG+1).
C
  150 IF(R.GE.RN(JG+1)) GO TO 152
      IF(R.GE.RN(JG)) GO TO 154
  151 JG=JG-1
      IF(R.LT.RN(JG))GO TO 151
      GO TO 153
  152 JG=JG+1
      IF(R.GE.RN(JG+1))GO TO 152
  153 NCH=1
  154 CONTINUE
C
C
      PHI=PHIN(IG-1,JG)
      R1=RN(JG)
      R2=RN(JG+1)
      Z1=ZN(IG)
      Z2=ZN(IG-1)
      IGN=IG
      JGN=JG
      RETURN
C
C     SOUTH Z
C
C     ASSUMING ZS(IIS) LESS THAN OR EQUAL TO Z LESS THAN ZERO.
C
    2 IF(INT.NE.0) GO TO 200
      DO 20 I=2,IIS
      IG=IIS-I+2
      IF(Z.LT.ZS(IG-1)) GO TO 203
   20 CONTINUE
C
C     ACCEPT IF ZS(IG) LESS THAN OR EQUAL TO Z LESS THAN ZS(IG-1).
C
      IF(Z.GE.ZS(IG)) GO TO 204
  201 IG=IG+1
  200 IF(Z.GE.ZS(IG-1))GO TO 202
      IF(Z.LT.ZS(IG))GO TO 201
      GO TO 203
  202 IG=IG-1
      IF(Z.GE.ZS(IG-1))GO TO 202
  203 NCH=1
  204 CONTINUE
C
C     SOUTH R
C
C     ASSUMING RS(1) LESS THAN OR EQUAL TO R LESS THAN OR EQUAL TO RS(JJ
C
      IF(R.EQ.RS(JJS)) JG=JJS-1
      IF(R.EQ.RS(JJS)) GO TO 253
      IF(INT.NE.0) GO TO 250
```


BEST AVAILABLE COPY

```
PROGRAM FARKOS                                02/01/78    PAGE  45

      DO 25 J=2,IJS
      JG=J-1
      IF(R.LT.RS(J)) GO TO 251
   25 CONTINUE
C
C     ACCEPT IF RS(JG) LESS THAN OR EQUAL TO R LESS THAN RS(JG+1).
C
  250 IF(R.GE.RS(JG+1)) GO TO 252
      IF(R.GE.RS(JG))GO TO 254
  251 JG=JG-1
      IF(R.LT.RS(JG))GO TO 251
      GO TO 253
  252 JG=JG+1
      IF(R.GE.RS(JG+1))GO TO 252
  253 NCH=1
  254 CONTINUE
C
      PHI=PHIS(IG-1,JG)
      R1=RS(JG)
      R2=RS(JG+1)
      Z1=ZS(IG)
      Z2=ZS(IG-1)
      IGS=IG
      JGS=JG
      RETURN
      END

                            45
```



```
PROGRAM FARKOS                                02/01/78    PAGE  46

      SUBROUTINE TRACK
      COMMON/T/X,Y,Z,R,XDOT,YDOT,ZDOT,PHI,PHIR,PHIZ,ERGO,DERG,C1,C2,E
      DERG=(PHI + XDOT**2 + YDOT**2 + ZDOT**2)/E-1. - ERGO
      ERGO=ERGO + DERG
      VMAX=ABS(XDOT) + ABS(YDOT) + ABS(ZDOT)
C
C STEP CONTROL
C
      IF(R.EQ.0.)  PHIX=0.
      IF(R.EQ.0.)  PHIY=0.
      IF(R.EQ.0.) GO TO 1
      PHIX=PHIR*X/R
      PHIY=PHIR*Y/R
    1 SS=AMIN1(C2, C1/VMAX)
C
      DT=SS/(ABS(PHIX) + ABS(PHIY) + ABS(PHIZ) + 1.E-6)
      DT=AMAX1(DT, .01/VMAX)
C THE FOLLOWING CARD IS FOR ZERO-POTENTIAL TESTS
      IF(PHIR.EQ.0..AND.PHIZ.EQ.0.) DT=C1/E/VMAX
      X=X+DT*(XDOT-.25*DT*PHIX)
      Y=Y+DT*(YDOT-.25*DT*PHIY)
      Z=Z+DT*(ZDOT-.25*DT*PHIZ)
      XDOT=XDOT-.5*DT*PHIX
      YDOT=YDOT-.5*DT*PHIY
      ZDOT=ZDOT-.5*DT*PHIZ
      RETURN
      END

                            46
```

```
      SUBROUTINE ORBIT
      COMMON/ORB/RDOT,ZDOT,AMS,NTIME,R1,R2,Z1,Z2,R,Z,PHI,RSAV,ZSAV
      DIMENSION TIME(6)
      M=61
      TOOM=3.3333E 33
      ROUND=1.E-12
      TROUND=10.*ROUND
      SDOT=2.*R*RDOT
      S=R**2
      S1=R1**2
      S2=R2**2
C     FOR DIAGNOSTIC OUTPUT,  SET   NOUT   GREATER THAN 0
      NOUT=0
      IF (NOUT.GT.0)
     1WRITE(M, 700) S,S1,S2,SDOT,AMS,Z1,Z2
700   FORMAT(   1X,23HS,S1,S2,SDOT,AMS,Z1,Z2=,1P3E25.15/12X,1P4E25.15)
C
      IF(S.GT.0.) GO TO 30
C
C CASE S=0 (AND S1=R1=0)
C
      NTIME=2
      IF(RDOT.EQ.0.) GO TO 40
      A=(ZDOT/RDOT)*R2
C
      IF(ZDOT.GT.0..AND.Z.GT.(Z2-A))  GO TO 10
      IF(ZDOT.LT.0..AND.Z.LT.(Z1-A))  GO TO 20
      Z=Z+A
      IF(ABS(Z-Z2).LT.TROUND)Z=Z2
      IF(ABS(Z-Z1).LT.TROUND)Z=Z1
      R=R2
      NTIME=0
      RETURN
C
10    R=(Z2-Z)/ZDOT*RDOT
      IF(ABS(R-R2).LT.TROUND)R=R2
      IF(ABS(R-R1).LT.TROUND)R=R1
      Z=Z2
      RETURN
C
20    R=(Z1-Z) /ZDOT*RDOT
      IF(ABS(R-R2).LT.TROUND)R=R2
      IF(ABS(R-R1).LT.TROUND)R=R1
      Z=Z1
      RETURN
C
30    IF(SDOT.NE.0..OR.AMS.NE.0.) GO TO 50
C
40    IF(ZDOT.GT.0.) Z=Z2
      IF(ZDOT.LT.0.) Z=Z1
      RETURN
```



```
C
50    NR=0
      H=S/(SDOT**2 + 4.*AMS)
      IF (NOUT.GT.0)
     1WRITE(M, 701) NR,H
701   FORMAT(   1X,5HNR,H=,I8,1PE25.15)
C
60    NR=NR+2
      IF(NR.GT.4) GO TO 30
C
C INTERSECTIONS ALONG S1-LINE
C
      IF(NR.EQ.2) Q=S1-S
      IF (NOUT.GT.0)
     1WRITE(M, 702) NR,Q
702   FORMAT(   1X,5HNR,Q=,I8,1PE25.15)
C
C INTERSECTIONS ALONG S2-LINE
C
      IF(NR.EQ.4) Q=S2-S
      IF (NOUT.GT.0)
     1WRITE(M,710) NR,Q
710   FORMAT(  15H SECOND   NR,Q=,I8, 1PE25.15)
      DISCRM=(SDOT*H)**2 + Q*H
      IF (NOUT.GT.0)
     1WRITE(M,703) NR,DISCRM
703   FORMAT(1X,10HNR,DISCRM=,I8,1PE25.15)
      TIME(NR)=TIME(NR-1)=TOOM
      IF (ABS(DISCRM).LT.ROUND) DISCRM=0.
      IF(DISCRM.LT.0.) GO TO 60
      TZ1=TZ2=-4*SDOT*2.
      IF(DISCRM.EQ.0.) GO TO 70
      IF(AMS.GT.0.) ROOT=SQRT(DISCRM)
      IF (NR.EQ.2.AND.AMS.EQ.0.) ROOT=.5*R1/RDOT
      IF (NR.EQ.4.AND.AMS.EQ.0.) ROOT=.5*R2/RDOT
      TZ1=TZ1-ROOT * 2.
      TZ2=TZ2+ROOT * 2.
      IF (NOUT.GT.0)
     1WRITE(M,704) NR,ROOT,TZ1,TZ2
704   FORMAT(1X,16HNR,ROOT,TZ1,TZ2=,I8,1P3E25.15)
      IF(ZDOT.EQ.0.) GO TO 60
C
70    ZROOT1=Z + ZDOT*TZ1
      ZROOT2=Z + ZDOT*TZ2
      IF (NOUT.GT.0)
     1WRITE(M,705) NR,TZ1,TZ2,ZROOT1,ZROOT2
705   FORMAT(1X,25HNR, TZ1,TZ2, ZROOT1,ZROOT2=,I5,1P4E25.15)
C
C ASSESS INTERSECTIONS FOR SIGNIFICANCE
C
      IF(ZROOT1.GE.Z1.AND.ZROOT1.LE.Z2) TIME(NR-1)=TZ1
```


BEST AVAILABLE COPY

```
      IF(ZROOT2.GE.Z1.AND.ZROOT2.LE.Z2) TIME(NR)=TZ2
      GO TO 60
C
80    TIME(NR-1)=TZ1
      TIME(NR)=TZ2
      GO TO 60
C
90    CONTINUE
      DO 100 NR=5,6
      TIME(NR)=TOOM
      IF(ZDOT.EQ.0.) GO TO 100
C
C S-INTERSECTION ALONG Z1-LINE
C
      IF(NR.EQ.5) Z12=Z1
C
C S-INTERSECTION ALONG Z2-LINE
C
      IF(NR.EQ.6) Z12=Z2
      TS=(Z12-Z)/ZDOT
      SROOT=S + SDOT*TS + TS**2/4./H
      IF (NOUT.GT.0)
     1WRITE(M,706) NR,Z12,TS,SROOT
706   FORMAT(1X,16HNR,Z12,TS,SROOT=,I8,1P3E25.15)
C
C ASSESS ROOT FOR SIGNIFICANCE
C
      IF(SROOT.GE.S1.AND.SROOT.LE.S2) TIME(NR)=TS
100   CONTINUE
      IF (NOUT.GT.0)
     1WRITE(M,707) (N,TIME(N),N=1,6)
707   FORMAT(1X,10HN,TIME(N)=,3(I8,1PE25.15)/11X,3(I8,E25.15))
C
C FIND SHORTEST SIGNIFICANT TIME
C
      NTIME=-1
      TIMIN=TOOM
      DO 200 NR=1,6
      IF(TIME(NR).EQ.TOOM) GO TO 200
      IF(TIME(NR).GT.ROUND.AND.TIME(NR).LT.TIMIN) NTIME=NR
      IF(TIME(NR).GT.ROUND.AND.TIME(NR).LT.TIMIN) TIMIN=TIME(NR)
200   CONTINUE
      IF (NOUT.GT.0)
     1WRITE(M,708) NTIME,TIMIN
708   FORMAT(1X,12HNTIME,TIMIN=,I8,1P2E25.15)
C
C ADVANCE TO APPROPRIATE END-POINT
C
      IF(NTIME.LT.0) RETURN
      Z=Z + ZDOT*TIMIN
      S=S + SDOT*TIMIN + .25*TIMIN**2/H
```



```
      SDOT=SDOT + .5*TIMIN/H
      R=0.
      IF(S.GT.ROUND) R=SQRT(S)
      IF (NOUT.GT.0)
     1WRITE(M,709) Z,S,SDOT,R
709   FORMAT(1X,11HZ,S,SDOT,R=,1P4E25.15)
      RSAV=R
      ZSAV=Z
      IF(NTIME.EQ.1.OR.NTIME.EQ.2) R=R1
      IF(NTIME.EQ.3.OR.NTIME.EQ.4) R=R2
      IF(NTIME.EQ.5) Z=Z1
      IF(NTIME.EQ.6) Z=Z2
      DR=R-RSAV
      DZ=Z-ZSAV
      IF(ABS(DR).GT.TROUND.AND.NTIME.LE.4) NTIME=-2
      IF(ABS(DZ).GT.TROUND.AND.NTIME.GT.4) NTIME=-3
      IF(ANS.EQ.0..OR.R.EQ.0.) RETURN
      ROOT=.5*SDOT/R
      RETURN
      END
```


BEST AVAILABLE COPY

REFERENCES

Brundin, C.L., Effects of charged particles on the motion of an earth satellite, AIAA J., 1, 2529, 1963.

De, B.R., and D.R. Criswell, Intense localized photoelectric charging in the lunar sunset terminator region. 1. Development of potentials and fields, J. Geophys. Res., 82, 999, 1977.

Fahleson, U., Plasma-vehicle interactions in space - some aspects of present knowledge and future development, in Photon and Particle Interactions with Surfaces in Space, edited by R.J.L. Grard, p. 563, D. Reidel, Dordrecht, Holland, 1973.

Fredricks, R.W., and F.L. Scarf, Observations of spacecraft charging effects in energetic plasma regions, in Photon and Particle Interactions with Surfaces in Space, edited by R.J.L. Grard, p. 277, D. Reidel, Dordrecht, Holland, 1973.

Grard, R.J.L., editor, Photon and Particle Interactions with Surfaces in Space, D. Reidel, Dordrecht, Holland, 1973.

Parker, L.W., Numerical methods for computing the density of a rarefied gas about a moving object, Allied Research Associates, Inc. Report AFCRL-64-193, 1964.

Parker, L.W., Theory of the external sheath structure and ion collection characteristics of a rocket-borne mass spectrometer, Mt. Auburn Research Associates, Inc. Report AFCRL-71-0105, 1970.

Parker, L.W., Computer solutions in electrostatic probe theory, Mt. Auburn Research Associates, Inc. Report AFAL-TR-72-222, 1973.

Parker, L.W., Computation of collisionless steady-state plasma flow past a charged disk, Lee W. Parker, Inc. Report NASA CR-144159, 1976a.

Parker, L.W., Theory of electron emission effects in symmetric probe and spacecraft sheaths, Lee W. Parker, Inc. Report AFGL-TR-76-0294, 1976b.

Parker, L.W., Calculation of sheath and wake structure about a pillbox-shaped spacecraft in a flowing plasma, in Proceedings of the Spacecraft Charging Technology Conference, edited by C.P. Pike and R.R. Lovell, p. 331, Joint Air Force-NASA Report AFGL-TR-77-0051 and NASA TMX-73537, 1977.

Parker, L.W., and E.C. Whipple, Jr., Theory of a satellite electrostatic probe, Ann. Phys., 44, 126, 1967.

Parker, L.W., and E.C. Whipple, Jr., Theory of spacecraft sheath structure, potential, and velocity effects on ion measurements by traps and mass spectrometers, J. Geophys. Res., 75, 4720, 1970.

Pelizzari, M., and D.R. Criswell, Patchy photoelectric charging of the lunar terminator terrain, EOS Trans. Amer. Geophys. Union, 58, 1180, 1977.

Pike, C.P., and R.R. Lovell, editors, Proceedings of the Spacecraft Charging Technology Conference, Joint Air Force-NASA Report AFGL-TR-77-0051 and NASA TMX-73537, 1977.

Rothwell, P.L., A.G. Rubin, and G.K. Yates, in Proceedings of the Spacecraft Charging Technology Conference, edited by C.P. Pike and R.R. Lovell, p. 389, Joint Air Force-NASA Report AFGL-TR-77-0051 and NASA TMX-73537, 1977.

Rosen, A., editor, Spacecraft Charging by Magnetospheric Plasmas, Vol. 47, Progress in Astronautics and Aeronautics, AIAA-MIT Press, 1976.

Rosenbauer, H.R., Possible effects of photoelectron emission on a low energy electron experiment, in Photon and Particle Interactions with Surfaces in Space, edited by R.J.L. Grard, p. 139, D. Reidel, Dordrecht, Holland, 1973.

Samir, U., and A.P. Willmore, The equilibrium potential of a spacecraft in the ionosphere, Planet. Space Sci., 14, 1131, 1966.

Schroder, H., Spherically symmetric model of the photoelectron sheath for moderately large plasma Debye lengths, in Photon and Particle Interactions with Surfaces in Space, edited by R.J.L. Grard, p. 51, D. Reidel, Dordrecht, Holland, 1973.

Soop, M., Report on photosheath calculations for the satellite GEOS, Planet. Space Sci., 20, 859, 1972.

Soop, M., Numerical calculations of the perturbation of an electric field around a spacecraft, in Photon and Particle Interactions with Surfaces in Space, edited by R.J.L. Grard, p. 127, D. Reidel, Dordrecht, Holland, 1973.

Whipple, E.C., Jr., The equilibrium electric potential of a body in the upper atmosphere and in interplanetary space, Ph.D. thesis, George Washington University, 1965. (Also NASA Tech. Note X-615-65-296, 1965).

Whipple, E.C., Jr., Theory of spherically symmetric photoelectron sheath: A thick sheath approximation and comparison with the ATS-6 observation of a potential barrier, J. Geophys. Res., 81, 601, 1976a.

Whipple, E.C., Jr., Observation of photoelectrons and secondary electrons reflected from a potential barrier in the vicinity of ATS-6, J. Geophys. Res., 81, 715, 1976b.

Whipple, E.C., Jr., and L.W. Parker, Effects of secondary electron emission on electron trap measurements in the magnetosphere and solar wind, J. Geophys. Res., 74, 5763, 1969.